

# Gestión y creación informatizada de exámenes

## AExamen



### TRABAJO DE FIN DE GRADO EN INGENIERÍA INFORMÁTICA

### CURSO 2018-2019

**Daniel Fernández Carnero**  
**Samuel Javier García Moreno**

*Director*  
**Antonio Sarasa Cabezuelo**

**Facultad de Informática**  
**Universidad Complutense de Madrid**

**Madrid, Junio de 2019**

# Índice

Índice de figuras	6
Resumen	9
Abstract	10
1 Introducción	11
1.1 Motivación	11
1.2 Objetivos del trabajo	12
1.3 Estructura de la memoria	12
1 Introduction	13
1.1 Motivation	13
1.2 Goals	14
1.3 Memory structure	14
2 Estado del arte	15
2.1 LMS	15
2.1.1 Classroom, Google	15
2.1.2 Moodle	16
2.1.3 Blackboard Learn	16
2.1.4 Edmodo	16
2.2 Generación de contenidos y evaluaciones	16
2.2.1 eXeLearning	16
2.2.2 Google Forms	17
3 Tecnología empleada	17
3.1 Tecnologías del Front-End	17
3.1.1 JavaScript (JS)	17
3.1.2 Bootstrap	17
3.1.3 JQuery	18
3.1.4 Ajax	18
3.1.5 TempusDominus	18
3.1.6 CSS	18
3.1.7 HTML	18
3.2 Tecnologías del Back-End	18
3.2.1 Apache	19
3.2.2 PHP	19
3.2.3 CakePHP	19
3.2.4 FPDF	19
3.2.5 PHPMailer	19
3.3 Persistencia de datos	19
3.3.1 MySQL	20
3.3.2 JSON	20

3.4 Otras tecnologías	20
4 Casos de uso	20
4.1 Diagrama de casos de uso	20
4.2 Casos de uso referidos al profesor	24
P-1 Generar exámenes	24
P-2 Login	25
P-3 Gestionar perfil propio	25
P-4 Solicitar acceso	25
P-5 Restablecer contraseña	26
P-6 Crear examen	26
P-7 Gestionar preguntas	27
P-8 Gestionar exámenes	27
4.3 Casos de uso referidos al coordinador	28
C-1 Gestionar profesores de su asignatura	28
C-2 Definir parámetros de examen	28
C-3 Gestionar preguntas	29
C-4 Gestionar exámenes	29
4.4 Casos de uso referidos al administrador	29
A-1 Gestionar profesores	29
A-2 Gestionar perfil propio	30
A-3 Eliminar preguntas	30
A-4 Eliminar exámenes	31
A-5 Gestionar asignaturas	31
A-6 Gestionar log y peticiones	31
5 Arquitectura de la aplicación	33
5.1. Arquitectura	33
5.2. El patrón modelo-vista-controlador	33
6 Modelo de datos	35
6.1 Modelo Entidad-Relación	35
6.2. Base de datos	36
6.2.1 Tabla administradores	37
6.2.2 Tabla asignaturas	37
6.2.3 Tabla exámenes	38
6.2.4 Tabla exámenes_historial	38
6.2.5 Tabla exam_preg	38
6.2.6 Tabla peticiones_registro	38
6.2.7 Tabla preguntas	39
6.2.8 Tabla preguntas_historial	39
6.2.9 Tabla profesores	39
6.2.10 Tabla prof_asig_coord	40

7 Diseño de la aplicación	41
7.1 Colores	41
7.2 Diseño de páginas	41
7.3 Funcionalidad	41
7.3.1 Diagrama de clases	41
7.3.2 Crear y editar examen	43
7.3.3 Generar examen	44
7.3.4 Crear preguntas	45
7.3.5 Definir parámetros	46
7.3.6 Panel de control	47
7.3.7 Página principal	49
7.3.8 Perfil propio	50
7.3.9 Asignaturas de profesor	51
7.3.10 Una asignatura	52
7.3.11 Gestionar profesores de una asignatura	54
7.3.12 Gestionar asignaturas y coordinadores	55
7.3.13 Gestionar profesores del sistema	56
7.3.14 Login	57
7.3.15 Restablecer contraseña	59
7.3.16 Registrarse	60
7.3.17 Exámenes	61
7.3.18 Detalles examen.	62
7.3.19 Borrar examen	64
7.3.20 Preguntas	65
7.3.21 Detalles pregunta	66
7.3.22 Editar pregunta	67
7.3.23 Borrar pregunta	68
8 Evaluación	70
8.1 Estructura de la encuesta	70
8.2 Resultados	74
8.3 Muestra	79
9 Conclusiones y trabajo futuro	80
9.1 Conclusiones	80
9.2 Aportaciones	80
9.2.1 Daniel Fernández Carnero	80
9.2.2 Samuel Javier García Moreno	81
9.3 Líneas de trabajo futuro	81
9 Conclusions and future work	82
9.1 Objectives	82
9.2 Contribution	82

9.2.1 Daniel Fernández Carnero	82
9.2.1 Samuel Javier García Moreno	83
9.3 Future work lines	83
Bibliografía	84

# Índice de figuras

1. Figura 1. Diagrama de casos de uso de profesor
2. Figura 2. Diagrama de casos de uso de coordinador
3. Figura 3. Diagrama de casos de uso de Administrador
4. Figura 4. Diagrama de los componentes de la aplicación
5. Figura 5. Diagrama entidad-relación de la base de datos
6. Figura 6. Relaciones entre las tablas de la base de datos
7. Figura 7. Diagrama de clases de la aplicación
8. Figura 8. Crear y editar examen
9. Figura 9. Bucle de preguntas de cada tema (View/CrearExamenes/index.ctp)
10. Figura 10. Generar examen
11. Figura 11. Función encargada de la creación del PDF de examen  
(Model\GenerarExamen.php)
12. Figura 12. Añadir pregunta
13. Figura 13. Función encargada de insertar en base de datos la pregunta creada  
(Model\Pregunta.php)
14. Figura 14. Definir parámetros de coordinador
15. Figura 15. Función encargada de añadir input de temas dinámicamente  
(webroot\js\definirParametrosExam.js)
16. Figura 16. Panel de Control
17. Figura 17. Parte de la vista de Panel de Control con botones de descarga, visualización, reinicio y eliminación del log (View\PanelesControles\index.ctp)
18. Figura 18. Página principal de la vista como administrador
19. Figura 19. Parte de la vista que comprueba si es administrador para mostrar otros enlaces (View\PaginasPrincipales\index.ctp)
20. Figura 20. Perfil propio, cambio de nombre
21. Figura 21. Modal de cambio de nombre en perfil propio (View\Perfiles\index.ctp)
22. Figura 22. Asignaturas de un profesor
23. Figura 23. Función del modelo Asignatura que carga de base de datos la información de las asignaturas correspondientes a un profesor (Model\Asignatura.php)
24. Figura 24. Asignatura Ingeniería del Software vista por un coordinador de esta
25. Figura 25. Parte de la vista que, de tratarse de un coordinador, muestra los enlaces a Profesores y a Parámetros de la asignatura (View\Asignaturas\una\_asignatura.ctp)
26. Figura 26. Añadir un profesor a una asignatura
27. Figura 27. Llamada AJAX que obtiene todos los profesores que no están asignados a la asignatura y los pinta en la tabla para su elección  
(webroot\js\profesoresDeUnaAsig.js)
28. Figura 28. Añadir coordinadores a una asignatura
29. Figura 29. Parte de la llamada AJAX que recibe la lista de profesores de una asignatura y los ordena para poder seleccionarlos para coordinar dicha asignatura  
(webroot\js\gestionarAsigAdmin.js)
30. Figura 30. Asignaturas que coordina un profesor y formulario para cambiarlas

31. Figura 31. Petición AJAX que obtiene las asignaturas correspondientes a un profesor y las imprime en el modal de la Figura 30 empezando por las que coordina seguidas de las que no (webroot\js\profesoresAdmin.js)
32. Figura 32. Login
33. Figura 33. código de Login
34. Figura 34. Restablecer contraseña
35. Figura 35. código de restablecer contraseña
36. Figura 36. Registrarse
37. Figura 37. código de registrarse
38. Figura 38. Exámenes
39. Figura 39. código de Exámenes
40. Figura 40. Detalles examen
41. Figura 41. código de detalles examen
42. Figura 42. Borrar examen
43. Figura 43. código de borrar examen
44. Figura 44. Preguntas
45. Figura 45. código de preguntas
46. Figura 46. Detalles pregunta
47. Figura 47. código de detalles pregunta
48. Figura 48. Editar pregunta
49. Figura 49. código de editar pregunta
50. Figura 50. Borrar pregunta
51. Figura 51. código de borrar pregunta
52. Figura 52. Formulario de evaluación (I)
53. Figura 53. Formulario de evaluación (II)
54. Figura 54. Formulario de evaluación (III)
55. Figura 55. Edad de los encuestados
56. Figura 56. Sexo de los encuestados
57. Figura 57. Respuestas a “¿Trabajas en la enseñanza o similares?”
58. Figura 58. Satisfacción con la primera tarea
59. Figura 59. Satisfacción con la segunda tarea
60. Figura 60. Satisfacción con la tercera tarea
61. Figura 61. Satisfacción con la cuarta tarea
62. Figura 62. Satisfacción con la quinta tarea
63. Figura 63. Diagrama con las valoraciones de las tareas más difíciles por los usuarios
64. Figura 64. Diagrama con valoraciones de usabilidad, colores y menús por los usuarios
65. Figura 65. Login
66. Figura 66. Restablecer contraseña
67. Figura 67. Página principal
68. Figura 68. Exámenes del profesor
69. Figura 69. Crear examen
70. Figura 70. Detalles de examen
71. Figura 71. Generar examen
72. Figura 72. Editar examen
73. Figura 73. Borrar examen

- 74. Figura 74. Asignaturas de un profesor
- 75. Figura 75. Asignatura
- 76. Figura 76. Preguntas de asignatura
- 77. Figura 77. Añadir pregunta
- 78. Figura 78. Detalles de pregunta
- 79. Figura 79. Editar pregunta
- 80. Figura 80. Borrar pregunta
- 81. Figura 81. Perfil propio
- 82. Figura 82. Asignatura
- 83. Figura 83. Profesores de una asignatura
- 84. Figura 84. Parámetros de exámenes
- 85. Figura 85. Asignaturas
- 86. Figura 86. Añadir coordinadores a una asignatura
- 87. Figura 87. Exámenes
- 88. Figura 88. Profesores del sistema
- 89. Figura 89. Panel de control
- 90. Figura 90. Perfil propio administrador
- 91. Figura 91. Registrarse
- 92. Figura 92. Instalación XAMPP (I)
- 93. Figura 93. Instalación XAMPP (II)
- 94. Figura 94. Acceso a archivo de configuración de Apache en XAMPP
- 95. Figura 95. httpd.conf
- 96. Figura 96. Directorio raíz del proyecto
- 97. Figura 97. PHPMyAdmin (I)
- 98. Figura 98. PHPMyAdmin (II)
- 99. Figura 99. /app/Config/database.php Valores por defecto de configuración de la base de datos
- 100. Figura 100. Pantalla inicial de la aplicación web



# Resumen

Este proyecto está orientado a ayudar a los profesores de universidad en la tarea de crear exámenes y pruebas de evaluación en asignaturas en las que la docencia se realiza de forma coordinada.

Se ha creado una aplicación que permite elaborar y guardar exámenes de diferentes asignaturas a partir de preguntas creadas por los profesores que imparten de forma coordinada las mismas. La aplicación ofrece diferentes servicios tales como la gestión de exámenes y preguntas, organización del profesorado y exportación de los exámenes en formato PDF. Está basada en PHP y bases de datos MySQL.

Palabras clave: *Aplicación web, Exámenes, Asignaturas, Profesores, Preguntas, PDF, PHP, MySQL.*

# Abstract

This project is focused on the teaching world of university, making exams creation and distribution as easy as possible, improving internal organization between teachers and coordinators within a subject.

An application has been created to prepare and save exams of different subjects from questions created by teachers who teach them in a coordinated way. The application offers different services such as exams and questions management, organization of the teaching staff and exams exportation in PDF format. It is based on PHP with MySQL databases.

Keywords: *Web application, Exams, Subjects, Teachers, Questions, PDF, PHP, MySQL.*

# 1 Introducción

En este capítulo se va a presentar la motivación por la cual se llevó a cabo este proyecto, los objetivos y la estructura del mismo.

## 1.1 Motivación

Actualmente la mayoría de instituciones gestionan su contenido docente de manera manual. Un caso particular es la creación de exámenes en asignaturas coordinadas. La gestión manual produce problemas de organización, retardos y pérdida de documentos e información de un curso para otro, y se generan confusiones en los profesores que imparten una asignatura por primera vez.

Estos problemas se solventarían creando un sistema que automatizara la creación, edición y preparación de los exámenes en las asignaturas universitarias coordinadas.

## 1.2 Objetivos del trabajo

El objetivo principal del proyecto consiste en crear una herramienta que facilite la gestión coordinada de las pruebas de evaluación de una asignatura por el profesorado que imparte la misma.

A continuación, se enuncian un conjunto de objetivos más específicos:

- La herramienta se creará como una aplicación web usable e intuitiva.
- Facilitará la gestión y creación de preguntas para la generación de pruebas de evaluación.
- Permitirá gestionar a los profesores de una asignatura coordinada, pudiendo añadirlos, eliminarlos o modificarlos.
- Permitirá asociar a un profesor el papel de coordinador de una asignatura.
- Guardará un registro histórico con todas las acciones llevadas a cabo por cada profesor.
- Generará exámenes con un formato configurable.

## 1.3 Estructura de la memoria

La memoria del proyecto se ha estructurado de la siguiente manera:

- Capítulo 1, Introducción: se introduce el Trabajo de fin de Grado, se explica la motivación que llevó a realizar el trabajo, los objetivos de este y la estructura de la memoria.
- Capítulo 2, Estado del arte: se analizan otras aplicaciones que implementan funcionalidades parecidas a las desarrolladas en el presente trabajo.
- Capítulo 3, Tecnología empleada: se describen las herramientas tecnológicas utilizadas para desarrollar la aplicación.
- Capítulo 4, Casos de uso: se muestran los casos de uso definidos para la aplicación, divididos por roles.
- Capítulo 5, Arquitectura de la aplicación: se describen los componentes del sistema y cómo se relacionan en la aplicación.
- Capítulo 6, Modelo de datos: describe el modelo entidad-relación desarrollado y la base de datos implementada.
- Capítulo 7, Diseño de la aplicación: se describe la implementación de las funcionalidades de la aplicación y su diseño.
- Capítulo 8, Evaluación: se describe la evaluación de la aplicación realizada sobre usuarios reales así como los resultados obtenidos.
- Capítulo 9, Conclusiones y trabajo futuro: se proponen un conjunto de conclusiones y líneas de trabajo futuro.
- Bibliografía: recoge todas las referencias hechas a lo largo del trabajo con enlace al contenido referenciado.
- Anexo: formado por una guía de uso y una guía de instalación.

# 1 Introduction

In this chapter we will present the motivation for which this project was carried out, its objectives and the structure used.

## 1.1 Motivation

Most institutions manage their teaching content manually. This leads on to problems like organization, delays and loss of documents and information from one course to another. This generates confusion in teachers who teach a subject for the first time, when agreeing on the weight of the subject's contents in the evaluation or when sharing information about the subject among teachers.

These problems would be solved by creating a system that would allow teachers and coordinators to create, edit and prepare exams for university in an organized and computerized way.

## 1.2 Goals

The main goal of the project is to make the management of the tests and methods of evaluation of a subject easier, allowing the creation of exams in a coordinated way for the teaching staff.

A set of more specific objectives is set out below:

- An easy to use and intuitive web application will be created.
- It will facilitate the management and creation of questions for generating exams.
- It will facilitate the management of different teachers who coordinate to teach a subject and generate exams for it.
- It will access to the teachers of a subject, being able to add more, eliminate some and define any as coordinator of that subject.
- It will save a historical record with all the actions carried out by each teacher.
- It will generate exams equal to the ones currently used in university.

## 1.3 Memory structure

The project report has been structured as follows:

- Chapter 1, Introduction: introduces the End of Degree Work, explains the motivation that led to carry out the work and the objectives of this and the structure of the report.

- Chapter 2, State of the art: other applications that implement functionalities similar to those developed in this work are analysed.
- Chapter 3 Technology used: the technological tools used to develop the application are described.
- Chapter 4, Use cases: The use cases present in the application are shown, divided by roles, together with the full use case diagram.
- Chapter 5, Application architecture: The application architecture is presented describing the components of the system and how they relate to each other.
- Chapter 6, Data model: describes the data model based on the entity-relationship model developed and the database implemented.
- Chapter 7, Application design: describes the design showing the implementation of the functionalities.
- Chapter 8, Evaluation: describes an evaluation of the application made on real users.
- Chapter 9, Conclusions and future work: a set of conclusions and future lines of work are proposed.
- Bibliography: includes all references made throughout the work listed with a link to the content in question.
- Finally, the annex: consisting of a user's guide and an installation guide.

## 2 Estado del arte

En este capítulo se explica de manera breve algunas herramientas disponibles en la actualidad que tienen funcionalidades parecidas a las utilizadas en este Trabajo de Fin de Grado. En este sentido, se van a analizar dos tipos de herramientas: LMS (Learning Management Systems), y herramientas de generación de contenidos y evaluaciones.

### 2.1 LMS

Los Learning Management Systems (LMS) son herramientas software empleadas para gestionar actividades y contenidos para la formación online.

#### 2.1.1 Google Classroom.

Es una plataforma online educativa de Google que permite gestionar todo lo relacionado con el aula de forma colaborativa. La herramienta permite crear y compartir documentos en diferentes formatos entre alumnos y profesores, elaborar tests de evaluación y entregas de trabajos para los alumnos. Esto se lleva a cabo de la siguiente manera: primero se crea un aula con alumnos, el profesor puede adjuntar diferentes tipos de ficheros al aula, puede crear alertas, avisos y crear encuestas. Para una agilización en el proceso de añadir alumnos a una clase, el profesor puede proporcionar a su alumnado un código que, introduciéndolo a la plataforma, se añade a ese alumno al aula establecida por el profesor. [1]

#### 2.1.2 Moodle

Moodle es un sistema de gestión de aprendizaje en línea. Permite a los educadores crear su sitio privado para compartir sus contenidos con los alumnos, por ellos tienen un sistema de diferentes roles, en el cual a cada rol se le permite unas ciertas acciones y se le prohíben otras. También permite personalizar el diseño de todo el sistema. Es posible crear temarios, foros de discusión, tareas con entregas en una fecha concreta, grupos de trabajo, evaluación de tests y creación de actividades colaborativas entre los alumnos. [2]

#### 2.1.3 Blackboard Learn

Blackboard es una aplicación de enseñanza, aprendizaje, creación de comunidades y uso compartido de conocimientos en línea. Permite la enseñanza en entorno síncrono y asíncrono. El entorno asíncrono consiste en una interacción en tiempo real entre profesores y alumnos donde se imparte la clase, se realiza un debate o se realiza una sesión de estudio conjunta. La versión asíncrona está presente fuera de la síncrona formando parte de esta: las entregas con fecha programada, los documentos de temario subidos por el profesor, las calificaciones del último test, etc. [3]

### 2.1.4 Edmodo

Edmodo es una plataforma educativa que hace las veces de red social entre el profesorado, los alumnos e incluso las familias de los alumnos. Permite la creación de un espacio virtual de comunicación con los alumnos donde intercambiar mensajes, documentos, calendarios, proponer tareas, actividades, gestionarlas, etc. Esto permite un feedback con el profesorado por parte de los alumnos. En Edmodo hay tres tipos de perfiles: los profesores, los estudiantes y los padres. Los profesores pueden crear grupos y administrarlos, estos grupos pueden englobar una materia o a una clase. Dentro de cada grupo se le permite realizar multitud de tareas, tales como crear eventos, subir archivos o calificar alumnos. Los alumnos podrán unirse a diversos grupos conociendo la clave de acceso a estos, también podrán consultar los archivos subidos por el profesor, ver clasificaciones, enviar archivos al profesor y participar los chats de los distintos grupos a los que pertenezca. Los padres tendrán acceso a la plataforma y podrán ver la información a la que tiene acceso su hijo. [4]

## 2.2 Generación de contenidos y evaluaciones

Las herramientas de generación de contenidos y evaluaciones son productos software que permiten desde la creación de material didáctico hasta pruebas de evaluación.

### 2.2.1 eXeLearning

Se trata de un programa modular, libre y abierto para crear contenidos educativos de manera fácil. Está desarrollado en Python y es multiplataforma, por lo que funciona tanto en Linux, como en Windows o MacOS. El aspecto de la página es fácilmente modificable, ya que se puede cambiar entre diferentes hojas de estilo. Permite crear contenidos didácticos en soportes informáticos variados (discos locales, para web, para la nube) sin necesidad de ser experto en lenguajes de marcado (HTML, XML o HTML5). Estos contenidos se incluyen con sencillez en diferentes LMS como Moodle. Con esta herramienta se pueden generar sitios web completos con contenido interactivo como preguntas, actividades, tests... Además ofrece la exportación de los contenidos creados a formatos ePub3, IMS o SCORM. [5]

### 2.2.2 Google Forms

Google Forms es una herramienta online de Google que permite crear formularios online. Estos formularios sirven desde para planificar eventos y realizar encuestas a hacer pruebas de evaluación y obtener los resultados estadísticamente según se vaya completando. Permite la distribución de los formularios a través de enlaces. Presenta la opción de autenticación obligatoria con cuenta Google de manera que se garantiza que exista únicamente una contestación por correo electrónico. Se pueden configurar los formularios de manera que después de enviar la respuesta, se pueda editar más adelante.



## 3 Tecnología empleada

En este capítulo se presentan las tecnologías empleadas en el proyecto. Para ello se van a agrupar en cuatro categorías: tecnologías del Front-End, tecnologías del Back-End, persistencia de datos y otras tecnologías.

### 3.1 Tecnologías del Front-End

A continuación se explican las tecnologías del lado del cliente utilizadas.

#### 3.1.1 JavaScript (JS)

Lenguaje de programación del lado del cliente que permite realizar actividades complejas en una página web. Permite controlar contenido multimedia, editar elementos del documento HTML de manera dinámica, usar variables dinámicas, reaccionar a eventos dentro de la página, etc. En la actualidad existen numerosos frameworks que usan como base este lenguaje por su versatilidad. [10]

#### 3.1.2 Bootstrap

Framework de diseño (JavaScript y CSS) desarrollado y liberado por Twitter con el objetivo de facilitar el diseño de páginas web responsives o adaptables. Utiliza elementos web para ayudar a que el diseño sea adaptable. Uno de los aspectos más destacables es su grid system, el cual permite maquetar el contenido por columnas añadiendo clases a los contenedores del HTML. [11]

#### 3.1.3 JQuery

Biblioteca multiplataforma libre y de código abierto de JavaScript que permite simplificar la interacción con los elementos del documento HTML (que de otra manera requerirían de más código JavaScript), el manejo de eventos, la creación de animaciones y la mejora de la interacción con AJAX en JavaScript. Además permite a los desarrolladores crear complementos de manera abstracta para crear interacciones, animaciones y efectos de alto nivel. [12]

#### 3.1.4 Ajax

Es una técnica de desarrollo web, que usa varias tecnologías en conjunto, tales como HTML, CSS, JavaScript, DOM y XSLT. Permite refrescar de forma asíncrona páginas web con información obtenida de APIs REST o de ficheros scripts, añadiendo y eliminando del árbol DOM diversos elementos y atributos dinámicamente. [13]

#### 3.1.5 TempusDominus

Biblioteca JavaScript, perteneciente a Bootstrap, que proporciona un calendario dinámico para seleccionar la fecha visualmente. [14]

### 3.1.6 CSS

Es un lenguaje de hoja de estilos que permite definir diferentes apariencias, y con ello un diseño visual, a distintos documentos confeccionados con lenguajes de marcado, tales como XML y HTML. Tiene una sintaxis simple, usando un conjunto de palabras en inglés para señalar diferentes propiedades de los elementos de los lenguajes de marcado. Para especificar a qué elemento quieren darle el estilo, usan los distintos atributos que poseen dichos elementos. Tiene múltiples ventajas, tales como facilitar la accesibilidad, ya que permiten que personas con algún tipo de discapacidad puedan acceder a la página; aportar consistencia del sitio, ya que gracias a CSS los estilos se comparten entre toda la página, lo que hace que la aplicación tenga un aspecto más uniforme. [15]

### 3.1.7 HTML

Es un lenguaje de marcado ampliamente usado para la confección de páginas web. Está compuesto de una serie de elementos que a su vez pueden tener distintos tipos de atributos. Junto con PHP, JavaScript y CSS forman el esquema general de cualquier aplicación web. Actualmente la mayoría de las páginas están confeccionadas en HTML. [16]

## 3.2 Tecnologías del Back-End

A continuación se explican las tecnologías del lado del servidor utilizadas.

### 3.2.1 Apache

Es un servidor web HTTP para diferentes plataformas, tales como Unix, Microsoft Windows, Macintosh, etc. Es de código abierto bajo licencia Apache 2.0 y desarrollado en lenguaje C. Este servidor tiene múltiples ventajas, tales como que es modular y extensible. [20]

### 3.2.2 PHP

Lenguaje script del lado del servidor de código abierto orientado al desarrollo web dinámico. El código PHP permite contener a su vez código HTML, de tal forma que introduce elementos dinámicos a las páginas web estáticas. El servidor que almacena un script PHP recibe una petición y procesa el script para servir al cliente una página web aparentemente estática, pero que se ha confeccionado dinámicamente con PHP. Puede generar archivos XML o XHTML y comunicarse con bases de datos en distintos formatos, aparte de bases de datos relacionales SQL. [7]

### 3.2.3 CakePHP

Es un marco de desarrollo (o framework) para PHP de código abierto. Se trata de una estructura que sirve de base en la creación de aplicaciones web haciendo uso de la arquitectura Modelo Vista Controlador (MVC). Permite generar código automáticamente, manejar peticiones, cookies y sesión, crear rutas personalizadas e interactuar con la base de datos mediante CRUD integrado. [22]

### 3.2.4 FPDF

Es una biblioteca gratuita y de licencia abierta escrita en PHP que permite crear archivos en formato PDF. Permite elegir una unidad de medida, el formato de página, los márgenes, establecer cabecera y pie de página personalizados. También permite adjuntar imágenes, cambiar los colores de los elementos e incrustar enlaces. Está desarrollado con orientación a objetos y el objeto `fpdf` es el encargado de almacenar la estructura del PDF mostrándolo cuando se llama a su función “output”. [8]

### 3.2.5 PHPMailer

Librería gratuita que facilita el envío de correo electrónico en PHP en comparación con la funcionalidad nativa “`mail()`”. Permite enviar correos electrónicos sin necesidad de un servidor de correo local (Windows no posee esta característica), acepta todos los formatos de correo típicos (a, copia, copia oculta, responder a). Además permite adjuntar archivos y enviar el cuerpo del correo formateado como HTML junto con una versión alternativa para clientes de correo que no lean HTML. [9]

## 3.3 Persistencia de datos

A continuación se explican las tecnologías de persistencia de datos utilizadas.

### 3.3.1 MySQL

Es un sistema de bases de datos relacionales desarrollado en los lenguajes C y C++. Está bajo una licencia dual, por una parte tiene la licencia de código abierto (GPL) y por otra está bajo la licencia de uso comercial gestionada por Oracle. Funciona sobre distintas plataformas, tales como Windows, GNU/Linux, Mac OS x, etc. [17]

### 3.3.2 JSON

Es un lenguaje de marcado que se usa como mecanismo de intercambio de datos. Es muy útil debido a su estilo compacto, a que se puede leer y escribir fácilmente por personas y a su vez los computadores lo interpretan y lo generan con rapidez. Su estructura está basada en claves/valor, esto facilita su transformación en objetos en diversos lenguajes que tienen esta misma estructura. [18]

## 3.4 Otras tecnologías

El desarrollo se ha llevado a cabo haciendo uso de Sublime Text [6] y Atom [23] como editores de texto y Git (a través de GitHub Desktop) para el control de las versiones del código y las subidas de este.

## 4 Casos de uso

En este capítulo se definen los casos de uso de la aplicación junto con los diagramas correspondientes.

### 4.1 Diagrama de casos de uso

En la aplicación desarrollada se han definido los siguientes actores:

**Profesor:** representa al usuario básico de la aplicación. Posee acceso a las asignaturas y la creación de exámenes a partir de preguntas de las mismas.

**Coordinador:** representa a profesores que coordinan una asignatura y por lo tanto a otros profesores. Además de lo que puede hacer un profesor, administra los aspectos internos de la asignatura.

**Administrador:** representa al usuario superior a los demás. Es capaz de dar acceso a la plataforma a los profesores, ascender profesores a coordinadores y acceder al panel de control de la aplicación.

En la figura 1 se muestra el diagrama de casos de uso referido al profesor. Tal como se puede observar un profesor puede crear exámenes, gestionar preguntas, editar su información personal o cambiar su contraseña.

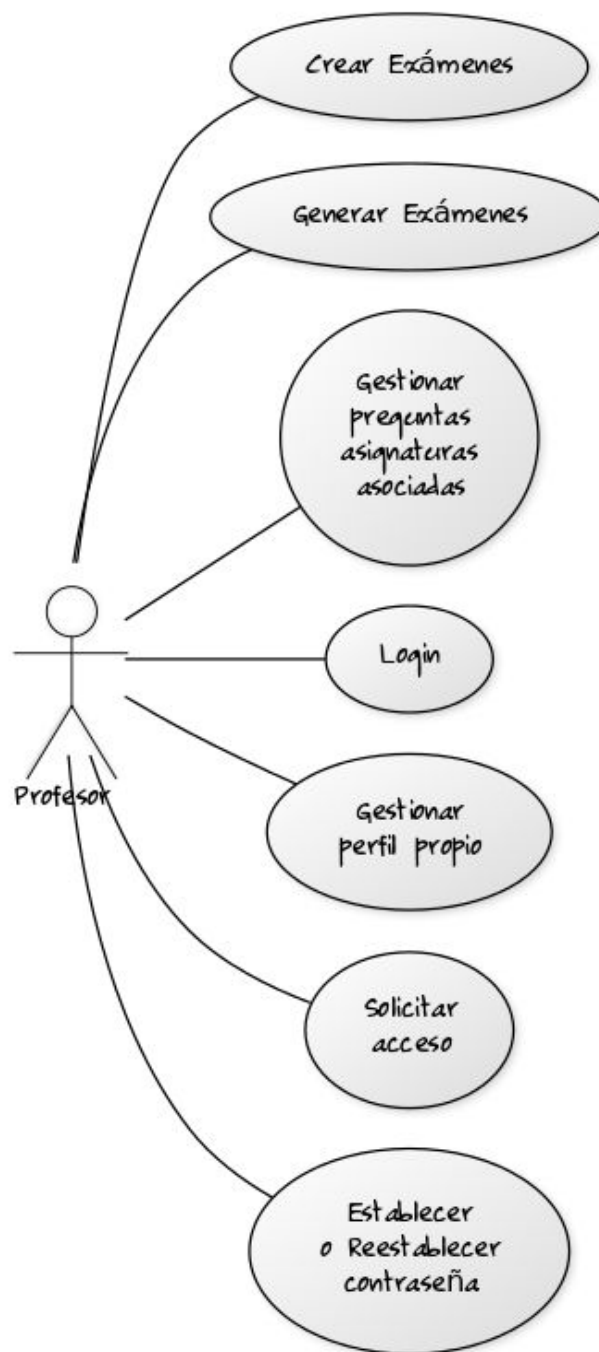


Figura 1. Diagrama de casos de uso de profesor

En la figura 2 se muestra el diagrama de casos de uso referido al coordinador. Tal como se puede observar un coordinador puede definir los parámetros de una asignatura en concreto y gestionar los profesores, exámenes y preguntas de su/s asignatura/s, además de todo lo que un profesor es capaz de hacer.

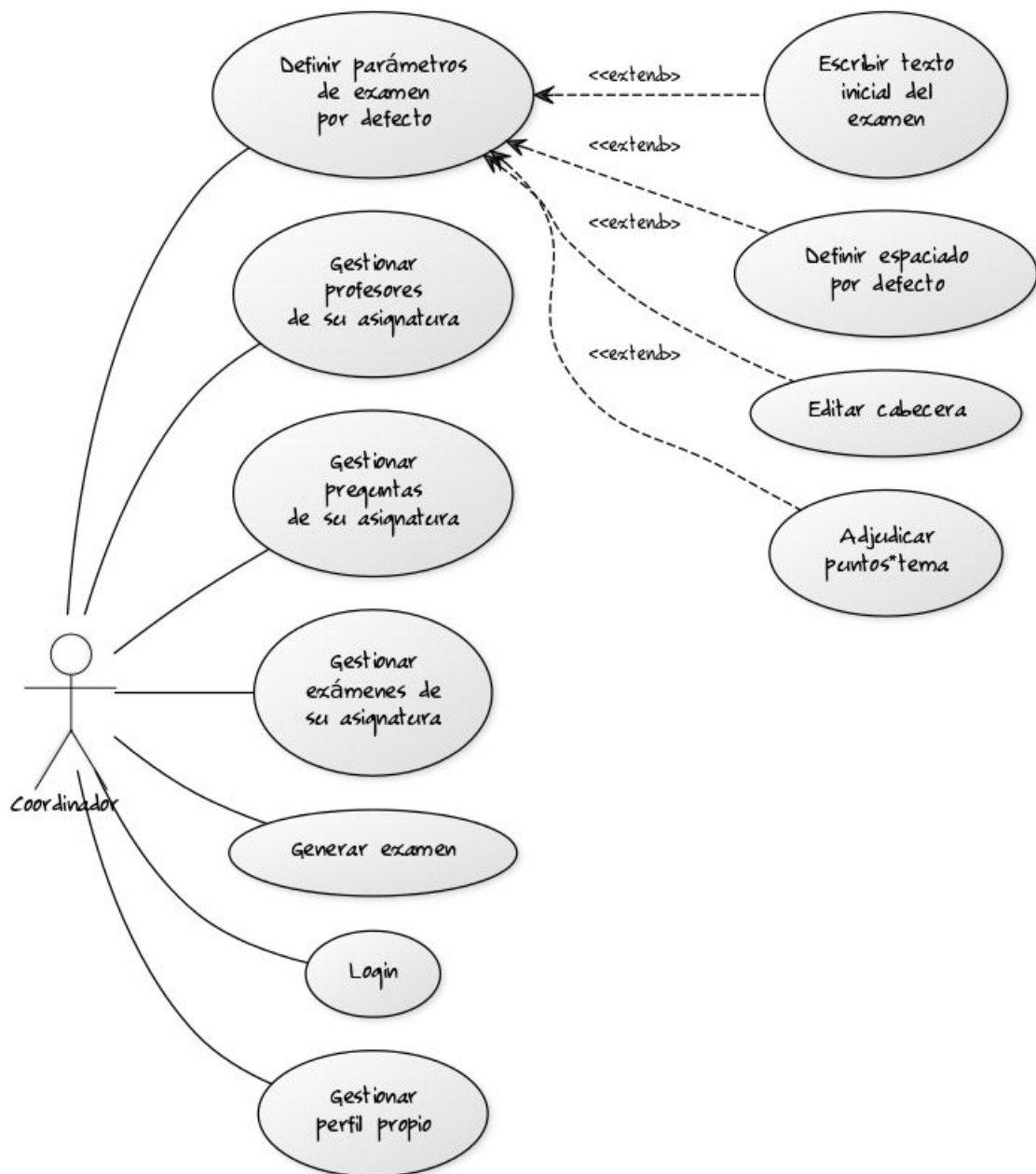


Figura 2. Diagrama de casos de uso de coordinador

En la figura 3 se muestra el diagrama de casos de uso referido al administrador. Tal como se puede observar un administrador puede gestionar los profesores y coordinadores de las asignaturas, eliminar exámenes y preguntas, gestionar el registro histórico (log) y aceptar o denegar las peticiones de registro entrantes al sistema.

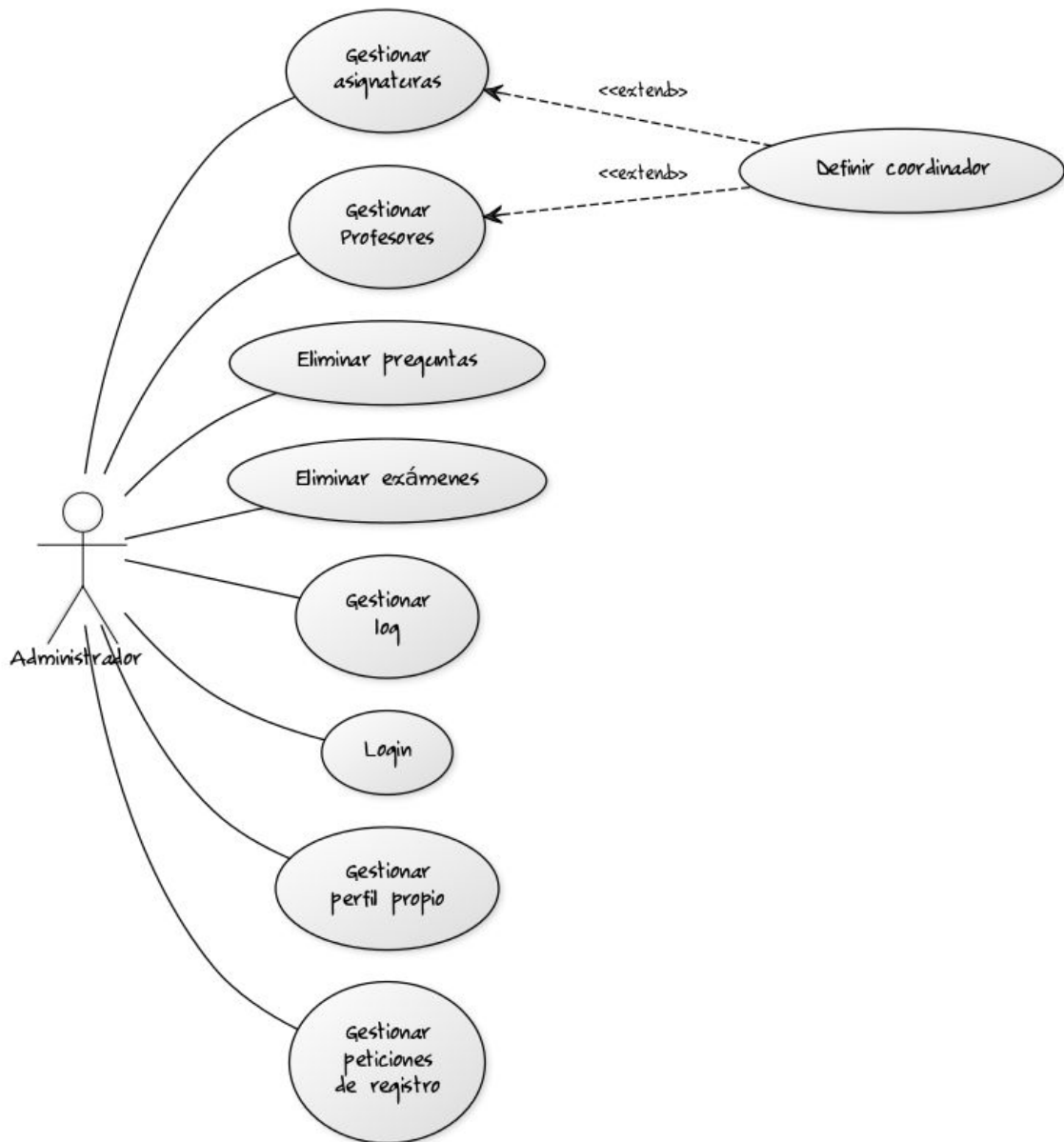


Figura 3. Diagrama de casos de uso de Administrador

Los casos de uso compartidos por todos los actores son: hacer login y gestionar el perfil propio. El login es el punto de inicio de casi todas las interacciones con la aplicación ya que es necesario para acceder a las asignaturas de cada actor. Respecto a gestionar el perfil propio, cada actor tiene acceso a sus datos y a modificarlos en todo momento.

En las siguientes secciones se presentarán en detalle los casos de uso agrupados por el tipo de actor en detalle. A continuación se explican de manera breve todas las acciones que puede llevar a cabo cada actor:

El actor profesor debe solicitar acceso a la aplicación la primera vez que quiera iniciar sesión. En caso de que se olvide de la contraseña de su cuenta podrá restablecerla desde

fuera de la aplicación. Una vez dentro de la aplicación es capaz de crear y generar exámenes, así como gestionar las preguntas de las asignaturas que imparte. Respecto a gestionar el perfil propio, el actor profesor puede modificar todos sus datos a excepción del correo con el que hace login.

El actor coordinador comparte con el profesor los casos de uso de gestionar el perfil propio y de generar exámenes. También comparte los casos de uso de gestionar preguntas y exámenes de las asignaturas que coordina, sólo que además puede eliminar cualquier pregunta sin importar el autor. Por otro lado, además de estos casos de uso compartidos con el profesor, el actor coordinador puede definir los parámetros de los exámenes y gestionar los profesores de las asignaturas que coordina. Los parámetros de examen que un coordinador puede editar son el texto inicial del examen, el espaciado por defecto entre las preguntas, la estructura de la cabecera y la cantidad de puntos adjudicados por cada tema de la asignatura.

El actor administrador comparte con los profesores y coordinadores el caso de uso de gestionar el perfil propio con la salvedad de poder editar su propio correo electrónico de acceso. En cuanto a los exámenes y preguntas, sólo puede verlos y borrarlos sin importar el autor. Puede gestionar las asignaturas y los profesores de la aplicación así como las solicitudes de acceso al mismo. Por último, tendrá acceso a un log en el cual se reflejan todas las acciones llevadas a cabo en la aplicación y podrá descargarlo, reiniciarlo y eliminarlo.

## 4.2 Casos de uso referidos al profesor

A continuación se muestran los casos de uso del actor profesor en detalle.

### P-1 Generar exámenes

Caso de uso	P-1: Generar exámenes
Precondiciones	Un profesor quiere imprimir un examen de los que hay creados en la asignatura que imparte.
Descripción	<p>El profesor elige uno de los exámenes creados y configura los siguientes aspectos del formato del examen:</p> <ol style="list-style-type: none"> <li>1) El espacio en blanco que habrá entre pregunta y pregunta.</li> <li>2) El cuatrimestre en el cual se hace el examen. Es un campo obligatorio.</li> <li>3) Los elementos que aparecerán en la cabecera del examen (grupo y letra del alumno, escudo de la facultad, que aparezca o no el campo DNI y la fecha del día del examen)</li> <li>4) Las indicaciones y consejos dirigidos a los alumnos acerca de la realización del examen.</li> </ol>



	<p>Cada uno de estos parámetros tendrá asignado un valor por defecto (menos el cuatrimestre), de manera que si el profesor no cambia estos valores, el examen se genera con los valores por defecto y el cuatrimestre seleccionado previamente.</p> <p>Después de configurar el examen pulsará sobre el botón “generar”</p>
--	---

## P-2 Login

Caso de uso	P-2: Login
Precondiciones	Un profesor quiere acceder a su cuenta en el sistema.
Descripción	El profesor accederá al sistema y le aparecerá una pantalla de login donde deberá introducir un correo y una contraseña. Tras rellenarlo pulsará sobre el botón de solicitud, se comprobarán los credenciales proporcionados, y en caso de ser correctos se permitirá el acceso al sistema. En caso de ser incorrectos, se mostrará un mensaje de error.

## P-3 Gestionar perfil propio

Caso de uso	P-3: Gestionar perfil propio
Precondiciones	Un profesor quiere modificar su nombre, apellidos o contraseña.
Descripción	El profesor iniciará sesión en el sistema, entrará en su perfil. Se mostrarán sus datos personales y al lado de cada uno de ellos aparecerá un icono que permitirá realizar su modificación. Después de realizar la modificación, se pulsará sobre un botón “aceptar” para hacer permanentes los cambios. En el caso de la contraseña, se le pedirá al profesor que introduzca dos veces la nueva contraseña con el objetivo de evitar posibles errores. Si quisiera modificar el correo electrónico, lo tendría que solicitar al administrador.

## P-4 Solicitar acceso

Caso de uso	P-4: Solicitar acceso
-------------	-----------------------

Precondiciones	Un profesor quiere acceder al sistema en el que no tiene cuenta.
Descripción	El profesor entrará en la ventana de login del sistema y pulsará en un enlace de registro. A continuación, aparecerá una ventana en la que el profesor deberá introducir sus datos personales, un correo electrónico, una contraseña y un texto de observaciones opcional. Una vez introducidos los datos, pulsará sobre el botón "registrarse". El registro deberá ser validado por un administrador. Una vez aceptada la solicitud, el usuario recibirá en su correo electrónico información del alta.

### P-5 Restablecer contraseña

Caso de uso	P-5: Restablecer contraseña
Precondiciones	Un profesor que posee una cuenta en el sistema quiere recuperar la contraseña para acceder a él.
Descripción	El profesor accederá a la página de login y pulsará sobre un enlace para recuperar la contraseña. A continuación aparecerá una pantalla donde se le pedirá que introduzca un correo electrónico. Una vez introducido, se pulsará sobre un botón de "aceptar" y el sistema comprobará que el correo electrónico se encuentra registrado. En caso de encontrarse registrado se enviará un correo electrónico que contiene un enlace. Si se pulsa sobre el enlace, se redirigirá a una ventana donde podrá introducir una nueva contraseña para la cuenta. En caso de no encontrarse registrado el correo electrónico introducido, entonces se le mostrará un mensaje de error.

### P-6 Crear examen

Caso de uso	P-6: Crear examen
Precondiciones	Un profesor desea crear un examen eligiendo las preguntas de cada tema hasta completar los puntos totales que tiene un examen
Descripción	El profesor entrará a una página donde podrá llevar a cabo la creación de un examen. Para crear un examen realizará los siguientes pasos:

	<p>1) Seleccionará o deseleccionará preguntas para incluir en el examen, respetando los puntos máximos establecidos por el coordinador para cada tema.</p> <p>2) Escogerá el título del examen.</p> <p>3) Pulsará en el botón de guardar.</p>
--	---

## P-7 Gestionar preguntas

Caso de uso	P-7: Gestionar preguntas
Precondiciones	Un profesor quiere ver las preguntas que hay en su asignatura, filtrar entre ellas y crear una propia.
Descripción	El profesor asignado a una asignatura podrá gestionar todas las preguntas de esta. Podrá filtrar entre las preguntas mediante una barra de búsqueda, así como buscar por autor de las preguntas. Podrá ordenar las preguntas alfabéticamente por nombre, autor/a y/o asignatura. Además podrán crear nuevas preguntas pulsando en “+” e introduciendo título, cuerpo y tema.

## P-8 Gestionar exámenes

Caso de uso	P-8: Gestionar exámenes
Precondiciones	Un profesor quiere ver todos los exámenes de las asignaturas a las que está asignado.
Descripción	El profesor podrá ver los exámenes de las asignaturas a las que está asignado. Podrá crear, editar y eliminar todos los exámenes de la asignatura que él haya creado. Para crear un examen tendrá que seleccionar de qué asignatura quiere crear el examen en el desplegable y una vez que haya seleccionado la asignatura deberá pulsar el botón con el símbolo “+”, que le redireccionará a la página de crear examen. Para editar un examen, se deberá pulsar en el botón con el símbolo del lápiz, y le redireccionará a la página de crear examen con los valores que tiene ese examen cargados, en esa pantalla podrá modificar lo que desee del examen y a continuación guardarlo. Por último, para borrar un examen deberá pulsar sobre el botón con el icono de la papelera del examen que desee borrar, tras esto se le mostrará un modal, en el que se le pedirá que confirme si desea borrar este examen, si confirma, se le mostrará otro modal con la misma pregunta, si

	confirma este último se borrará el examen. A su vez podrá filtrar los exámenes mediante búsquedas por título rellenando la barra del buscador con las palabras que quiera filtrar, así como ordenar alfabéticamente por título, autor/a, fecha de modificación y fecha de creación mediante los diversos menús desplegables que se encuentran en la parte superior de la tabla.
--	---

### 4.3 Casos de uso referidos al coordinador

A continuación se muestran los casos de uso del actor coordinador en detalle.

#### C-1 Gestionar profesores de su asignatura

Caso de uso	C-1: Gestionar profesores de su asignatura
Precondiciones	Un coordinador quiere añadir/quitar profesores a la asignatura que coordina.
Descripción	El coordinador accederá a la pantalla de la asignatura que coordina y pulsará sobre un botón para gestionar profesores de la asignatura. A continuación se le desplegará una pantalla en la cual aparecerán los profesores que tiene asociados a su asignatura. Podrá quitar de la asignatura a cualquier profesor asociado a esta. Podrá añadir nuevos profesores a su asignatura pulsando el botón de añadir profesor.

#### C-2 Definir parámetros de examen

Caso de uso	C-2: Definir parámetros de examen
Precondiciones	Un coordinador quiere establecer los parámetros de un examen para que los profesores puedan crear exámenes basándose en estos.
Descripción	El coordinador de una asignatura podrá editar una serie de parámetros que aparecerán predefinidos cuando se crea un examen de una asignatura. Se podrán elegir los puntos por tema que se deben asignar obligatoriamente a las preguntas (este parámetro no podrá ser modificado por el profesor mientras crea un examen), definir el espaciado entre preguntas y un texto inicial

	que tendrá el examen para indicar consejos y normas a los estudiantes. Estos dos últimos parámetros sí podrán ser modificados por un profesor cuando vaya a generar un examen.
--	--

### C-3 Gestionar preguntas

Caso de uso	C-3: Gestionar preguntas
Precondiciones	Un coordinador quiere ver las preguntas que hay en su asignatura y filtrar entre ellas, crear una propia o eliminar cualquiera de ellas.
Descripción	El coordinador realizará las mismas acciones que un profesor respecto a las preguntas, pero además podrá eliminar cualquier pregunta sin importar el autor de esta.

### C-4 Gestionar exámenes

Caso de uso	C-4: Gestionar exámenes
Precondiciones	Un coordinador quiere ver, editar o borrar exámenes de las asignaturas a las que está asignado.
Descripción	El coordinador realizará las mismas acciones que un profesor en esta vista, además podrá eliminar cualquier examen, sea del autor que sea.

## 4.4 Casos de uso referidos al administrador

A continuación se muestran los casos de uso del actor administrador en detalle.

### A-1 Gestionar profesores

Caso de uso	A-1: Gestionar profesores
Precondiciones	El administrador quiere modificar los datos de un profesor, eliminarlo de la plataforma o elegir qué asignaturas coordina.
Descripción	El administrador accederá a una ventana donde se mostrarán todos los profesores que hay dados de alta en el sistema:

	<p>1) Para modificar un profesor, el administrador pulsará sobre un botón de modificación que le muestra una ventana para modificar los datos y guardarlos.</p> <p>2) Si se quiere borrar un profesor del sistema, se pulsará sobre un botón de borrar y se pedirá que confirme la acción. No se permitirá dejar una asignatura sin coordinadores en ningún momento.</p> <p>3) Si quiere añadir un profesor al sistema, tendrá dos opciones. Si el profesor ya ha solicitado el acceso al sistema, podrá confirmar o denegar la solicitud. En cambio si no ha solicitado el acceso al sistema, el administrador invitará al profesor introduciendo el email del mismo y pulsará en enviar.</p> <p>4) Si se quiere ascender a un profesor al rol de coordinador se seleccionarán las asignaturas que coordinará este profesor y se confirmará. En ningún momento se podrá dejar una asignatura sin coordinador.</p>
--	--

## A-2 Gestionar perfil propio

Caso de uso	A-2: Gestionar perfil propio
Precondiciones	El administrador quiere modificar su nombre, apellidos, contraseña y/o correo electrónico.
Descripción	El administrador entrará en su perfil propio desde la barra de navegación. Podrá cambiar lo mismo que un profesor con la diferencia de que el administrador también podrá modificar su correo electrónico

## A-3 Eliminar preguntas

Caso de uso	A-3: Eliminar preguntas
Precondiciones	El administrador quiere borrar preguntas de una asignatura.
Descripción	El administrador entrará a la vista de preguntas de una asignatura y pulsará sobre el botón con forma de papelera y confirmará la eliminación de la pregunta en cuestión.

## A-4 Eliminar exámenes

Caso de uso	A-3: Eliminar preguntas
Precondiciones	El administrador quiere eliminar algún examen.
Descripción	El administrador buscará el examen que quiera eliminar en la vista de exámenes. Lo eliminará haciendo click sobre la papelera, después de confirmar que quiere eliminarlo.

## A-5 Gestionar asignaturas

Caso de uso	A-5: Gestionar asignaturas
Precondiciones	El administrador desea gestionar las asignaturas de la aplicación.
Descripción	El administrador puede ver las asignaturas dadas de alta en la plataforma. A su vez visualiza el número de profesores asignados a cada asignatura y el/los coordinador/es. Puede realizar búsquedas, para filtrar los resultados por título, siglas de las asignaturas o nombres de los coordinadores. También se pueden definir los coordinadores de la asignatura, para ello pulsará un botón y se le mostrará un pop-up con una tabla con todos los profesores del sistema, apareciendo primero los profesores que están seleccionados como coordinadores. Cada profesor tiene un checkbox, seleccionado si es un coordinador de la asignatura y no seleccionado en caso contrario. Se pueden editar los checkboxes para seleccionar qué profesor coordinará la asignatura y cuál no.

## A-6 Gestionar log y peticiones

Caso de uso	A-6: Gestionar log y peticiones
Precondiciones	El administrador desea descargar o reiniciar el log del sistema. También aceptar o denegar peticiones de registro de nuevos profesores
Descripción	El administrador podrá consultar, descargar, reiniciar y eliminar el log que se genera en el sistema. También podrá ver la última vez que se modificó. En cuanto a las peticiones, podrá aceptar o denegar las peticiones de acceso al sistema pulsando sobre los

	botones correspondientes. Hasta que estas peticiones no se acepten, el usuario no podrá acceder a la aplicación. En las peticiones se mostrará información del usuario que solicita el acceso.
--	--



## 5 Arquitectura de la aplicación

En este capítulo se explicará cómo se organiza la aplicación y qué componentes la forman. Para ello se presentará primero el diagrama de la arquitectura completa y después de explicará el patrón utilizado para el desarrollo del proyecto.

### 5.1. Arquitectura

Se ha implementado una aplicación web cuya arquitectura completa se muestra en la figura 4.

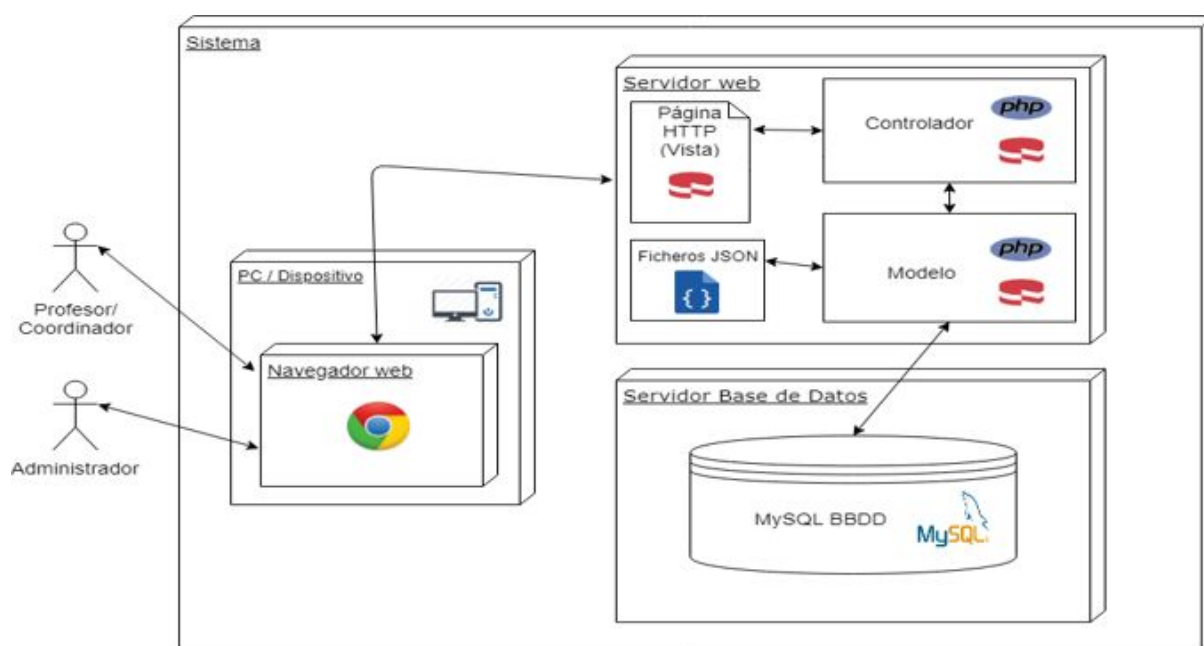


Figura 4. Diagrama de los componentes de la aplicación

### 5.2. El patrón modelo-vista-controlador

Se ha desarrollado una aplicación web que utiliza un patrón modelo vista controlador (MVC) (Figura 4). Este patrón consiste en tener diferenciado por una parte la vista, la cual no realiza ninguna tarea de procesamiento, únicamente muestra la información; el modelo, el cual interactúa con la base de datos, solicitando información; y por último el controlador que procesa la información que proporciona el modelo para enviarla a la vista.

Las vistas se componen de archivos (.ctp) que usan los lenguajes HTML, PHP, CSS y JavaScript para implementarlas. Estas vistas son accesibles desde un navegador web, el cual interpreta los diferentes ficheros, los muestra y permite al usuario interactuar con esta.

En ella se llevan a cabo peticiones al controlador solicitando la información necesaria para realizar la acción que solicita el usuario.

Los controladores están conformados por ficheros JavaScript y PHP, reciben peticiones por parte de las vistas, solicitan información y los datos necesarios al modelo, y una vez obtenidos se procesan de la forma que se requiera para poder mostrarlos a través de la vista. A su vez, en caso de que los datos no sean correctos o se produzca un error a la hora de tratarlos, se informa también a la vista.

El modelo está formado por ficheros PHP. Estos reciben peticiones por parte del controlador y solicitan esta información a la base de datos MySQL o a un fichero JSON. Una vez que se obtiene la información, esta se envía al controlador. En caso de que se produzca un fallo, se notifica al controlador de lo ocurrido para que este actúe.

La vista y el controlador se comunican entre ellos mediante las peticiones GET y POST del protocolo HTTP y mediante la variable SESSION de PHP. Las peticiones GET y POST se usan para comunicar solicitudes, que lleva a cabo el usuario en la vista, al controlador y del controlador a la vista. La variable SESSION sirve para almacenar datos que se van a usar a lo largo de la sesión que ha iniciado un usuario.

El controlador se comunica con el modelo mediante AJAX, enviando peticiones POST a través del protocolo XMLHttpRequest. El modelo devuelve los datos como respuesta a estas peticiones. Para acceder a los datos se usa la utilidad de acceso CRUD de CakePHP para conectar con la base de datos MySQL y realizar consultas. Las credenciales de acceso se obtienen accediendo mediante PHP a un fichero JSON donde están almacenadas.

## 6 Modelo de datos

En este capítulo se desarrollará el modelo de datos utilizado en el proyecto. Para ello en primer lugar se explicará el modelo entidad-relación pensado y después se desarrollará la base de datos implementada a partir de este modelo.

Esta aplicación se apoya en el modelo de datos para almacenar toda la información necesaria para el correcto funcionamiento de la aplicación. Se ha utilizado una base de datos relacional implementada en MySQL y un archivo JSON. La base de datos relacional se usa para almacenar toda la información perteneciente a la aplicación (exámenes, preguntas, profesores, etc.). El fichero JSON se usa para almacenar las credenciales de autenticación para mandar correos electrónicos automáticos con el fin de aislar estas credenciales de la base de datos, de forma que únicamente se almacenen en este fichero.

## 6.1 Modelo Entidad-Relación

En la figura 5 se muestra el modelo entidad-relación que se ha definido para realizar la persistencia de la información del sistema desarrollado

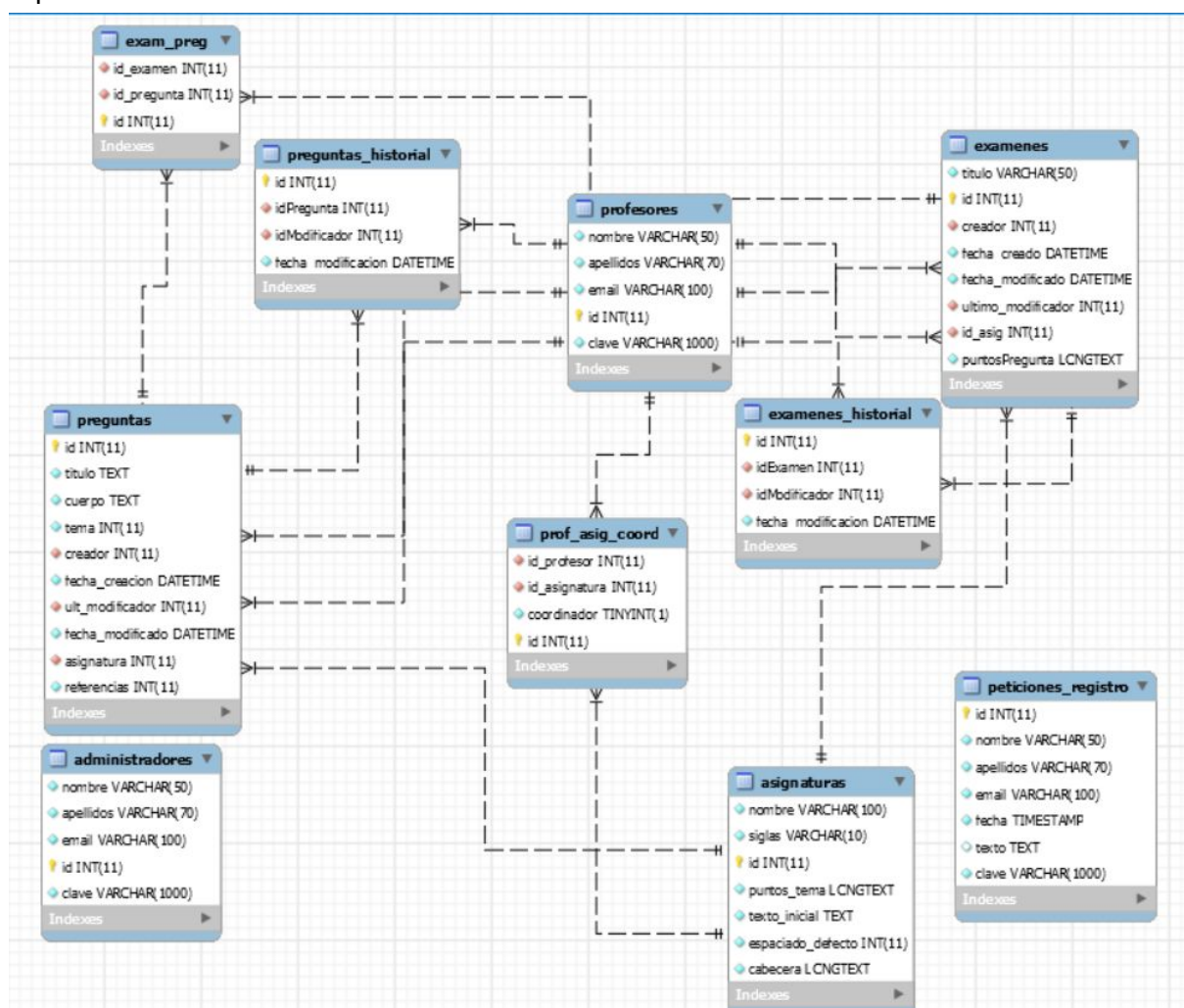


Figura 5. Diagrama entidad-relación de la base de datos

El modelo de datos contempla a los profesores como figura principal. Estos están conectados con las preguntas, asignaturas y exámenes ya que guardan relación entre ellos. Por otra parte los administradores y las peticiones de registro están aislados ya que son independientes del resto de información.

## 6.2. Base de datos

El modelo entidad-relación planteado se ha implementado usando una la base de datos relacional. En la figura 6 se muestra la estructura de la base de datos. Tal como se aprecia está formada por 10 tablas. A continuación se describe cada una de las tablas.

Consta de 8 tablas relacionadas entre sí de forma binaria y ternaria y 2 tablas independientes de las demás y entre ellas. A continuación se muestra un diagrama de las tablas y campos de la base de datos (Figura 6).

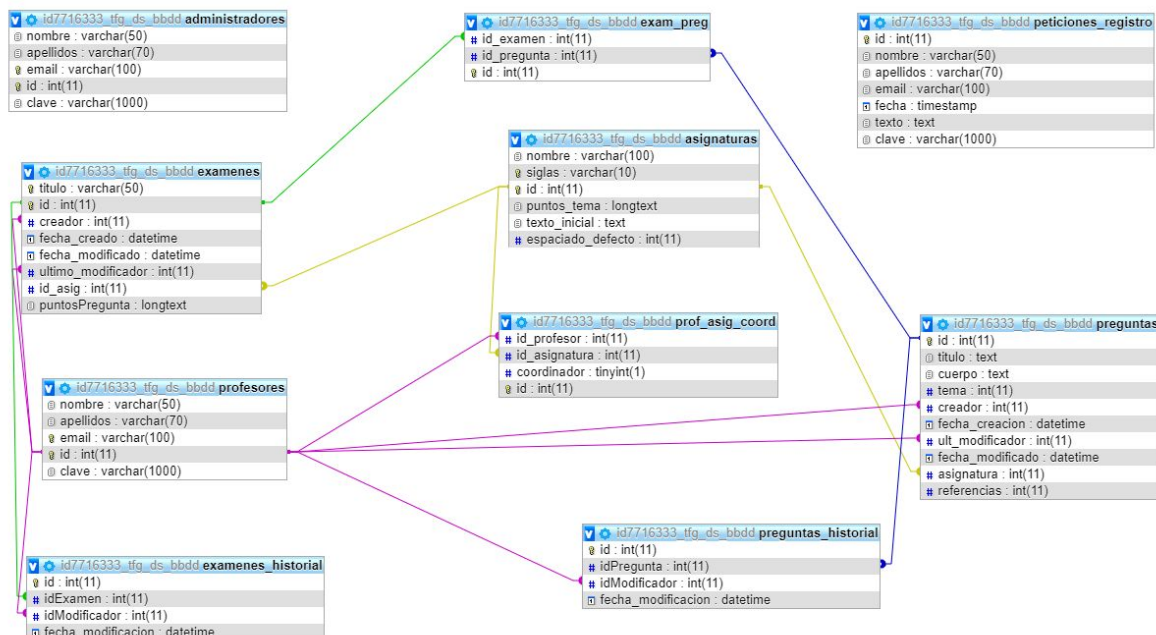


Figura 6. Relaciones entre las tablas de la base de datos

### 6.2.1 Tabla administradores

En esta tabla se guarda la información de los administradores del sistema. Los campos de la tabla son los siguientes:

- id: Identificador único del administrador.
- nombre: Nombre del administrador.
- apellidos: Apellidos del administrador
- email: Email del administrador. Es único, por lo que no puede haber dos administradores con el mismo email. Necesario en el login para acceder al sistema.
- clave: Clave hasheada del administrador. Necesaria en el login para acceder al sistema.

### 6.2.2 Tabla asignaturas

En esta tabla se guarda la información de las asignaturas gestionadas por el sistema. Los campos de la tabla son los siguientes:

- id: Identificador único de la asignatura.
- nombre: Nombre de la asignatura.
- siglas: Siglas de la asignatura.
- puntos\_tema: Texto en formato JSON que guarda los puntos que hay definidos por cada tema en los exámenes de la asignatura.
- texto\_inicial: Texto que hay definido por defecto en los exámenes de la asignatura.
- espaciado\_defecto: Entero que indica el espaciado que va a haber entre las preguntas de un examen de la asignatura

### 6.2.3 Tabla exámenes

En esta tabla se guarda la información sobre los exámenes que están creados. Los campos de la tabla son los siguientes:

- id: Identificador único del examen.
- titulo: Título del examen.
- creador: Identificador del profesor que ha creado el examen.
- fecha\_creado: Fecha en la cual el examen fue creado.
- fecha\_modificado: Fecha de la última modificación del examen.
- ultimo\_modificador: Identificador del último profesor que ha modificado el examen.
- id\_asig: Identificador de la asignatura a la cual pertenece el examen.
- puntosPregunta: Texto en formato JSON en el cual se define por cada tema, que preguntas hay y cuantos puntos tiene cada una. La pregunta está representada con el identificador de la pregunta.

### 6.2.4 Tabla examenes\_historial

En esta tabla se guarda información histórica de los exámenes creados con el sistema tales como: cuando se modificó un examen, de qué examen se trata y qué profesor lo modificó. Los campos de la tabla son los siguientes:

- id: Identificador único de la modificación.
- idExamen: Identificador único del examen modificado.
- idModificador: Identificador del profesor que lo modificó.
- fecha\_modificacion: fecha en la cual se modificó.

### 6.2.5 Tabla exam\_preg

En esta tabla se representan qué preguntas están en qué exámenes. Los campos de la tabla son los siguientes:

- id: Identificador único.
- id\_examen: Identificador del examen.
- id\_pregunta: Identificador de la pregunta.

### 6.2.6 Tabla peticiones\_registro

En esta tabla se almacenan las peticiones de los usuarios que aún no tienen acceso al sistema. Los campos de la tabla son los siguientes:

- id: Identificador único.
- nombre: Nombre del solicitante.
- apellidos: Apellidos del solicitante.
- email: Email del solicitante. En caso de que la petición la acepte el administrador, será el correo que tendrá que usar para acceder al sistema.
- fecha: Fecha en la cual se envió la petición.
- texto: Texto opcional que el usuario que hace la petición puede rellenar para que el administrador lo lea.
- clave: Clave con la cual el usuario quiere acceder al sistema. En caso de que al administrador acepte la petición, será la que use para acceder al sistema.

### 6.2.7 Tabla preguntas

En esta tabla está almacenada la información de todas las preguntas que se han introducido en el del sistema con su información. Los campos de la tabla son los siguientes:

- id: Identificador único de la pregunta.
- titulo: Título de la pregunta.
- cuerpo: Cuerpo de la pregunta.
- tema: Tema al que pertenece la pregunta.
- creador: Identificador del profesor que creó la pregunta.
- fecha\_creacion: Fecha en la cual se creó la pregunta.
- ult\_modificador: Identificador del último profesor que creo la pregunta.
- fecha\_modificado: Fecha de la última modificación de la pregunta.
- asignatura: Asignatura a la cual pertenece la pregunta.
- referencias: Número de veces que se usa la pregunta en el sistema.

### 6.2.8 Tabla preguntas\_historial

En esta tabla se almacena información histórica sobre cuando se modificó una pregunta, indicando de qué pregunta se trata y qué profesor la modificó. Los campos de la tabla son los siguientes:

- id: Identificador único.
- idPregunta: Identificador de la pregunta modificada.
- idModificador: Identificador del profesor que la modificó.
- fecha\_modificacion: Fecha en la que se modificó.

### 6.2.9 Tabla profesores

En esta tabla se almacena la información de todos los profesores del sistema con todos sus datos. Los campos de la tabla son los siguientes:

- id: Identificador único del profesor.
- nombre: Nombre del profesor.
- apellidos: Apellidos del profesor.
- email: Email del profesor con el cual se accede al sistema.
- clave: Clave del profesor con la cual accede al sistema. Se almacena hasheada para evitar problemas de seguridad.

### 6.2.10 Tabla prof\_asig\_coord

En esta tabla se relacionan los profesores con las asignaturas que imparten, indicando si son coordinadores de estas. Los campos de la tabla son los siguientes:

- id: Identificador único.
- id\_profesor: Identificador de un profesor.
- id\_asignatura: Identificador de una asignatura.
- coordinador: Booleano que indica si el profesor indica si es o no es coordinador de la asignatura.

## 7 Diseño de la aplicación

En este capítulo se exponen los colores usados en la aplicación, el diseño de las páginas y las funcionalidades en detalle.

### 7.1 Colores

Se ha elegido un color azul oscuro (#345) como color de fondo y un color blanco para la pantalla donde se muestra la información y donde se llevan a cabo prácticamente todas las acciones. Esto se ha debido principalmente a centrar la atención del usuario. Se ha establecido para los botones un color azul (#007bff), el cual se corresponde con el color predeterminado de Bootstrap [11] (conocido como primary). Mientras que los menús se han diseñado con una tonalidad semejante al tono azul oscuro del fondo (#2C3E50), para que destaquen sobre la pantalla de color claro donde se sitúan.

### 7.2 Diseño de páginas

Un buen diseño de las páginas es fundamental para que el usuario no se sienta confuso a la hora de utilizar una aplicación web. Por ello se han diseñado las páginas de esta aplicación de forma que muestre qué acción realiza cada botón y menú, reduciendo la probabilidad de que un usuario se sienta desorientado cuando utiliza la aplicación.

### 7.3 Funcionalidad

A continuación se explicarán las funcionalidades de la aplicación y cómo se han implementado.

#### 7.3.1 Diagrama de clases

En la figura 7 se muestra el diagrama de clases completo de la aplicación con las relaciones entre estas.



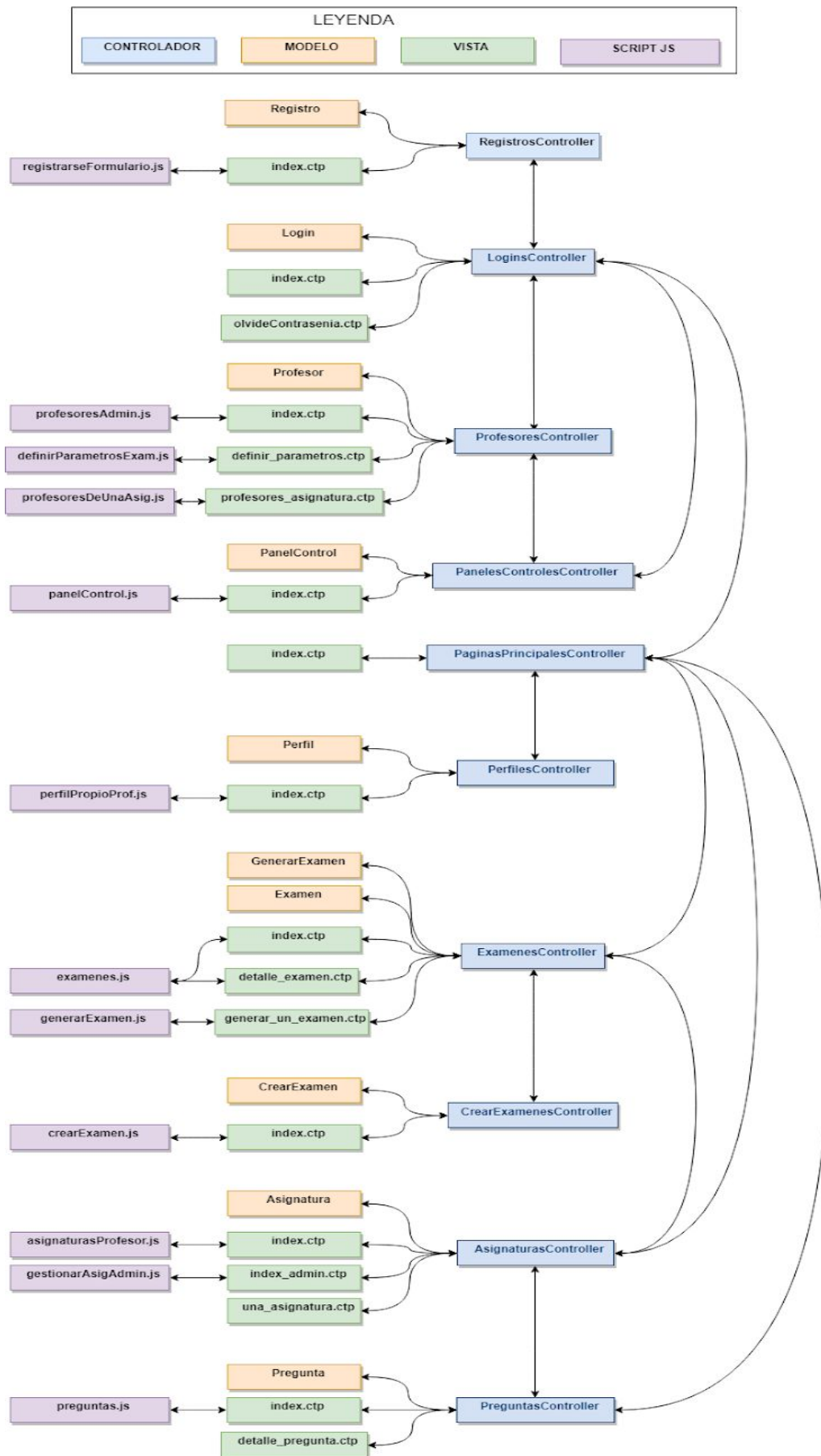


Figura 7. Diagrama de clases de la aplicación.

### 7.3.2 Crear y editar examen

En esta funcionalidad, en la pantalla de exámenes de una asignatura, un profesor quiere añadir un nuevo examen. Para ello pulsa en añadir examen y le aparece la pantalla de crear examen (Figura 8). Como resultado, se cargan desde la base de datos los temas, los puntos de cada tema, las preguntas de cada tema (Figura 9) y los puntos de cada pregunta. El profesor pulsa en los botones “+” que aparecen en el lado derecho del tema y, mediante JavaScript, se muestra un modal donde se muestran las preguntas de ese tema. Estas preguntas se cargan desde la base de datos al controlador y se pasan a la vista, donde se recorren todas mostrando el tema al que pertenecen, el título y el cuerpo de esta. Una vez que ha seleccionado una, se le muestra en la pantalla y puede definir cuántos puntos vale la pregunta sin poder sobrepasar el límite establecido para el tema. Esta comprobación se realiza mediante el uso de JQuery. Una vez que se han definido todas las preguntas, se establece un nombre para el examen y a continuación se puede guardar.



Figura 8. Crear y editar examen

```

echo'<div class="row" id="preguntasTema".$i.">';
if ($preguntasTema) {
    foreach ($preguntasTema as $pregunta) {
        $datos = $datosPreguntasSesion["tema".$i][$pregunta['id']];
        // $datos = cargaUnicaPregunta($pregunta['id']);
        echo '<div class="col-12 preguntaTema".$i.'" puntos="'.$pregunta['puntos'].'" id="'.$pregunta['id'].'">
            <b><span class="tituloPreguntas">'.$datos['titulo'].'</span></b>
            <br>
            <span class="cuerpoPreguntas">'.$datos['cuerpo'].'
            </span><br>
            <div class="row botonesPregunta">
                <div class="col-1 puntosPregunta" id="puntosPregunta'.$pregunta['id'].'">'.
                '<a class="fas fa-chevron-left" id="menosPuntosPregunta'.$pregunta['id'].'" asignatura= "'.$_GET["idAsignatura"].'" href="#"></a>
                <span class="puntos">'.$pregunta['puntos'].'</span><span> p </span>'.
                '<a class="fas fa-chevron-right" id="masPuntosPregunta'.$pregunta['id'].'" asignatura= "'.$_GET["idAsignatura"].'" href="#"></a>'.
                '</div>'.
                <div>'.
                '<a class="fas fa-times boton-eliminar" pregunta="'.$pregunta['id'].'" id="boton-eliminar" asignatura= "'.$_GET["idAsignatura"].'" href="#"></a>'.
                '</div>'.
                </div>';
    }
}
echo'</div>';

```

Figura 9. Bucle de preguntas de cada tema (View/CrearExamenes/index.ctp)

### 7.3.3 Generar examen

Cuando un profesor quiere generar un archivo PDF de un examen, accede a la pantalla donde se sitúan todos los exámenes de una asignatura y pulsa en el botón generar de cualquiera de ellos. Entonces se le muestra la pantalla en la que puede definir los parámetros del examen (Figura 10). En ella se cargan desde la base de datos algunos parámetros que vienen definidos por defecto (que ha definido por previamente el coordinador de dicha asignatura) y otros en blanco ya que no han sido definidos por defecto. No obstante, todos los parámetros los puede modificar el profesor. El parámetro “Cuatrimestre” no viene definido por defecto, y mediante JavaScript, se deshabilita el botón “Generar”, hasta que no se selecciona una opción. Una vez que se han definido los parámetros deseados, se pulsa en el botón “Generar”, que pasa los parámetros y valores del examen a la librería FPDF, la cual genera el examen en formato PDF respetando márgenes (Figura 11).

Figura 10. Generar examen

```

public function construirPDF ($postInfo, $preguntasExamen) {
    $pdf = new PDF();
    $pdf->AddPage();
    $pdf->SetFont('Arial', 'B', 11);
    $pdf->AliasNbPages();
    $pdf->cabecera();
    $pdf->SetTitle($_SESSION['nombreExamenGenerado'].".pdf");
    $pdf->SetFont('Arial', 'B', 11);
    $pdf->MultiCell(0, 5, $postInfo['pautas'], 1, 'L');
    $pdf->Ln(10);
    if ($preguntasExamen) {
        $i=1;
        foreach ($preguntasExamen as $pregunta) {
            $pdf->Cell(5, 5, $i." ", "", 0, 'R');
            $pdf->MultiCell(0, 5, "(".$pregunta['puntos']." puntos ".utf8_decode($pregunta['pregunta']));
            if ($postInfo['espaciado'] == 2) {
                $pdf->Ln(30);
            } else if ($postInfo['espaciado'] == 10) {
                $pdf->Ln(70);
            } else {
                $pdf->Ln(200);
            }
            $i++;
        }
    } else {
        $_SESSION['error_no_existen_preguntas'] = true;
        header('Location: generarExamen.php?examen='.$_SESSION['nombreExamenGenerado']);
    }
    $pdf->Output("I", $_SESSION['nombreExamenGenerado'].".pdf");
}

```

Figura 11. Función encargada de la creación del PDF de examen  
(Model\GenerarExamen.php)

### 7.3.4 Crear preguntas

En la pantalla de preguntas, un profesor puede añadir preguntas a la asignatura. Al pulsar el botón de añadir, mediante JavaScript se muestra un pop-up con tres campos (Figura 12). El primero de ellos es el título de la pregunta, el segundo el cuerpo de la pregunta y el último el número de tema al que pertenece la pregunta. Mientras alguno de los tres esté vacío, se bloquea el botón “Insertar” con JavaScript para impedir que se inserte una pregunta incompleta. Una vez el formulario ha sido enviado, en el modelo Pregunta se inserta la pregunta en la base de datos (Figura 13).

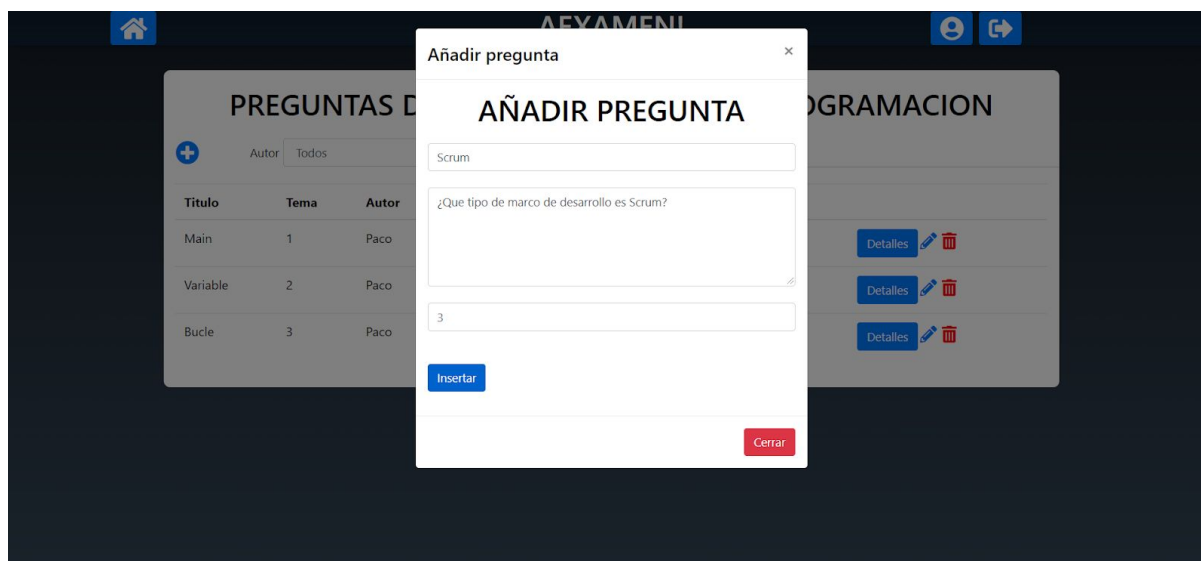


Figura 12. Añadir pregunta

```
function añadirPregunta($titulo,$cuerpo,$tema){
    $funciona=false;
    $date = date('Y-m-d H:i:s', time());
    $sql = "INSERT INTO `preguntas`(`titulo`, `cuerpo`, `tema`, `creador`, `fecha_creacion`, `ult_modificador`, `fecha_modificado`, `asignatura`)
    VALUES ('".$titulo."','".$cuerpo."','".$tema."','".$_SESSION['id']."','".$date."','".$_SESSION['id']."','".$date."','".$_SESSION['idAsignatura']."')";
    $consulta = $this->query($sql);

    $sqlNuevoID="SELECT id FROM `preguntas` ORDER BY `preguntas`.`id` DESC LIMIT 1";
    $consulta = $this->query($sqlNuevoID);
    $idPreguntaNueva = $consulta[0]['preguntas']['id'];

    $sql = "INSERT INTO `preguntas_historial`(`id`, `idPregunta`, `idModificador`, `fecha_modificacion`)
    VALUES ('".$_SESSION['id']."','".$idPreguntaNueva."','".$_SESSION['id']."','".$date.")";
    $this->query($sql);

    $sql = "SELECT `nombre` FROM `asignaturas` WHERE id='".$_SESSION['idAsignatura']";
    $consulta = $this->query($sql);
    $fila=$consulta[0]['asignaturas'];

    $funciona=true;
    //Escribimos en el log que usuario ha añadido la pregunta y a que asignatura
    $log = '['.date("d/m/Y - H:i:s").'] : ".USER --> id ".$_SESSION['id']. " - ".$_SESSION['apellidos'].', '.$_SESSION['nombre'].
    " | ACTION --> ".$_SESSION['email']. " creo una nueva pregunta en la asignatura ".$_fila['nombre'].PHP_EOL.
    "-----".PHP_EOL;
    file_put_contents('./log/log_AExamen.log', utf8_decode($log), FILE_APPEND);
    echo $funciona;
}
```

Figura 13. Función encargada de insertar en base de datos la pregunta creada (Model\Pregunta.php)

### 7.3.5 Definir parámetros

En esta página (Figura 14) el administrador de una asignatura define los parámetros que han de tener por defectos los exámenes de dicha asignatura. Entre los parámetros están el número de temas con la cantidad de puntos que ha de tener cada uno, así como el total de puntos que ha de tener el examen. En caso de que el total no coincida con la suma de los puntos de los temas, mediante JavaScript se muestra un error por pantalla y se bloquea el botón de guardar, para impedir que el usuario guarde datos incorrectos. Cuando se añaden o se quitan temas, de forma dinámica (mediante JQuery en la figura 15), se añaden o eliminan del árbol DOM los campos input de los temas correspondientes. También se define el espaciado entre preguntas, en el cual se puede elegir entre tres opciones (pequeño, mediano y grande o de página completa). Por último, el coordinador puede definir el texto inicial que estará en la cabecera del examen. Una vez que todo esto se haya

completado, se enviará el formulario pulsando sobre el botón “Guardar” y se insertarán los parámetros en la base de datos para la asignatura correspondiente. Si por algún casual el coordinador quiere deshacer los cambios que ha hecho antes de guardar puede pulsar sobre Restablecer valores.

Figura 14. Definir parámetros de coordinador

```
//añade temas de forma dinamica segun al formulario segun se van añadiendo
$('.numTemasForm').bind('keyup mouseup',function() {
    let contadorTemas = 0;
    $('.puntosTemaForm').each(function() {
        contadorTemas += 1;
    });
    //añadimos un tema si el numero de temas se incrementa
    if ($(this).val()>contadorTemas) {
        let dif = $(this).val()-contadorTemas;
        for (var i = 0; i < dif; i++) {
            $('#filaPuntosPorTema').append('<div id="div_tema'+(contadorTemas+1+i)+'" class="form-group col-4">'+
            '<label>Tema '+ (contadorTemas+1+i)+'</label>'+
            '<input type="number" class="form-control puntosTemaForm" id="tema'+(contadorTemas+1+i)+'" value="0" min="0">'+
            '</div>');
        }
    }
    //eliminamos un tema si el numero de temas se incrementa
    else if ($(this).val()<contadorTemas) {
        while($(this).val()!==contadorTemas){
            $('#div_tema'+contadorTemas).remove();
            contadorTemas--;
        }
    }
});
```

Figura 15. Función encargada de añadir input de temas dinámicamente  
(webroot\js\definirParametrosExam.js)

### 7.3.6 Panel de control

El acceso a esta vista está restringido a los administradores, se controla mediante la comprobación de diversas variables de sesión de PHP en el controlador, en caso de que el

usuario no tenga el rol del administrador se redirige a la página principal. En la vista del panel de control (Figura 16) se muestran diversas acciones que se pueden realizar sobre el log y las peticiones de registro que hay en la aplicación.

Respecto al log, está almacenado en un archivo en el servidor (log\_AExamen.log) en el cual se escribe mediante PHP cada vez que un usuario hace login, crea una pregunta, crea o genera un examen y cierra sesión, entre otras acciones. Si el usuario pulsa sobre el botón “Eliminar”, se muestra un modal mediante JavaScript pidiendo confirmación, en caso de que se confirme se borra el fichero que almacena el log usando PHP. También se puede ver de forma online el contenido del fichero y descargarlo en formato “.log”. Para descargar el archivo log se hace una llamada a “downloadLog.php” con el parámetro “file” indicando el nombre del archivo correspondiente. Para obtener la última vez que se modificó el archivo se hace uso de la función “filemtime” [21] de PHP (Figura 17), la cual obtiene el momento de la última modificación de un archivo. Además se hace uso de la función `formatoDateTime` la cual nos devuelve formateada la fecha y hora indicando cuántos segundos/minutos han pasado desde dicha fecha (hasta que pasen 24h, a partir de ahí se presenta la fecha estándar) de manera que sea más cómodo de leer para el usuario.

En cuanto a las peticiones de registro, el administrador podrá aceptarlas o denegarlas. Sea cual sea la acción llevada a cabo, se le mandará un email al solicitante usando la librería PHPMailer, la cual recibe por parámetro un HTML que se envía directamente al solicitante.

Para descargar el archivo log se hace una llamada a la función `descargarLog` con el parámetro `$_REQUEST['file']`, el cual contiene la ruta al fichero log.

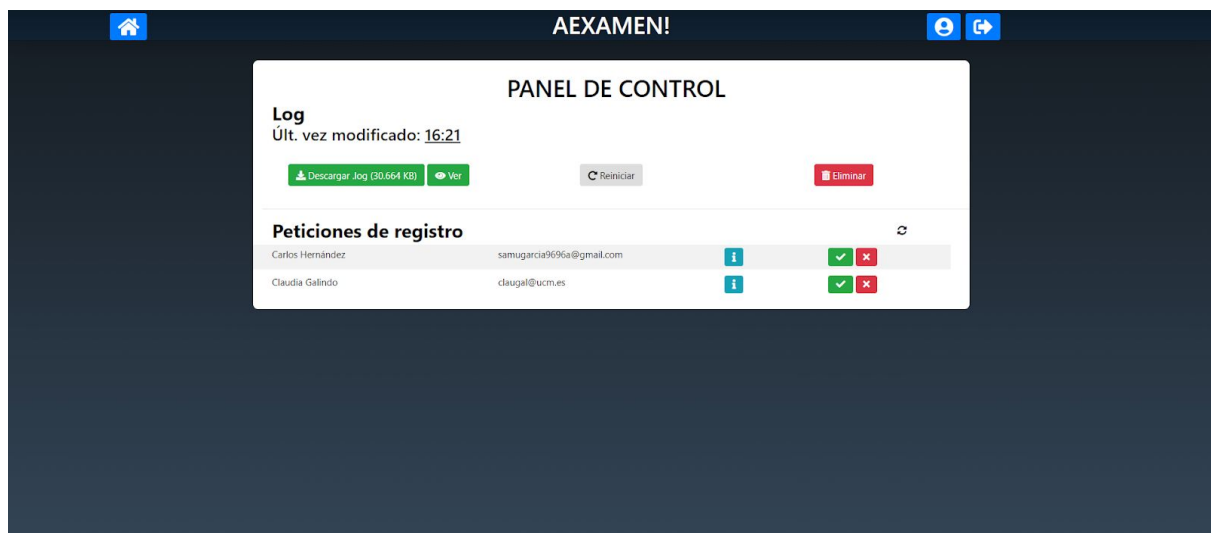


Figura 16. Panel de Control



```

<!-- LOG -->
<div class="row filaPanelControl">
  <div class="col-lg-12">
    <h2><b>Log</b></h2>
  </div>
  <div class="col-lg-12">
    <h3>Últ. vez modificado: <u><?php echo $ultimaModificacion; ?></u></h3>
  </div>
  <br><br><br>
  <div class="col-lg-4 centrar">
    <a href="/panelescontroles/descargarlog?file=log_AExamen.log">
      <button class="btn btn-success" id="descargarLog">
        <i class="fas fa-download"></i> Descargar .log (<?php echo round(floatval(filesize('./log/log_AExamen.log'))/(1024)),3). ' KB'; ?>)
      </button>
    </a>
    <a target="_blank" href="/log/log_AExamen.log"><button class="btn btn-success"><i class="fas fa-eye"></i> Ver</button></a>
  </div>
  <div class="col-lg-4 centrar">
    <button class="btn" id="reiniciarLog"><i class="fas fa-redo-alt"></i> Reiniciar</button>
  </div>
  <div class="col-lg-4 centrar">
    <button class="btn btn-danger" id="eliminarLog"><i class="fas fa-trash-alt"></i> Eliminar</button>
  </div>
</div>

```

Figura 17. Parte de la vista de Panel de Control con botones de descarga, visualización, reinicio y eliminación del log (View\PanelesControles\index.ctp)

### 7.3.7 Página principal

Las funcionalidades de esta vista dependen del rol del usuario que accede a esta (iniciando sesión). Si comprueba que se trata de profesores y coordinadores (mediante la variable de sesión establecida en el login) se limita a exponer links a Asignaturas y a Exámenes. Para administradores añade Profesores y Panel de control (Figura 19). Esta vista es la primera que se ve nada más iniciar sesión (Figura 18), y por lo tanto la primera que contiene el acceso a Perfil propio desde el menú superior. Los botones de acceso a las vistas contienen una animación hecha en CSS3 la cual desvela más información de la vista destino si se hace click sobre estos.

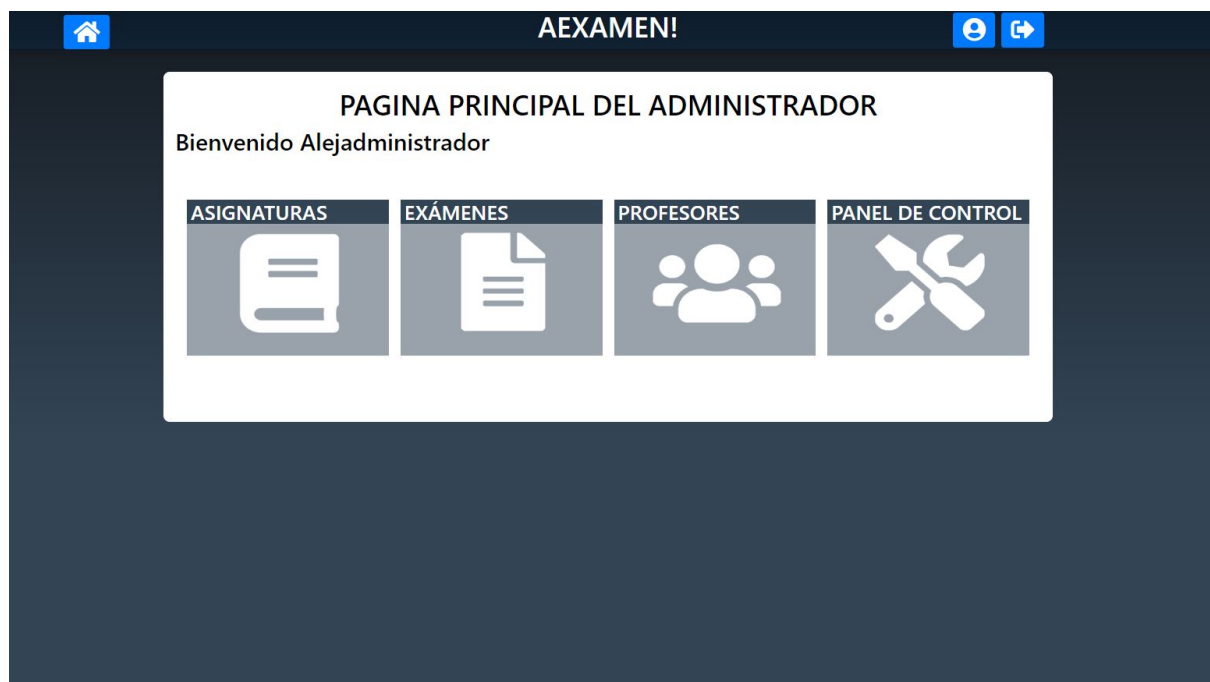




Figura 18. Página principal de la vista como administrador

```
<?php
if($_SESSION['administrador']){
?>
<div class="col-xl-3 col-lg-4 col-md-6 col-sm-3 portfolio-item filter-app">
  <h3>PROFESORES</h3>
  <a href="/profesores/index">
    
    <h3>PROFESORES</h3>
    <div class="details">
      <h4>PROFESORES</h4>
      <span>Gestionar profesores</span>
    </div>
  </a>
</div>

<div class="col-xl-3 col-lg-4 col-md-6 col-sm-3 portfolio-item filter-app">
  <h3>PANEL DE CONTROL</h3>
  <a href="/panelescontroles/index">
    
    <h3>PANEL DE CONTROL</h3>
    <div class="details">
      <h4>PANEL DE CONTROL</h4>
      <span>Ir al panel de control</span>
    </div>
  </a>
</div>
```

Figura 19. Parte de la vista que comprueba si es administrador para mostrar otros enlaces (View\PaginasPrincipales\index.ctp)

### 7.3.8 Perfil propio

Esta vista carga en el controlador (a partir de una llamada al modelo) el nombre, apellidos y correo electrónico del usuario que ha iniciado sesión y añade la funcionalidad de cambiar estos datos. Además, si se trata de un administrador, podrá cambiar el correo electrónico con el que accede al sistema. El acceso a esta vista está presente en todas las páginas (siempre y cuando se haya iniciado sesión) al estar situado en el menú de la barra superior. Para cambiar cualquier dato se ha de pulsar sobre el botón correspondiente, rellenar el modal y confirmar (no dejará confirmar a no ser que teclee o rellene el campo pedido para evitar guardar información errónea) (Figura 20 y 21).

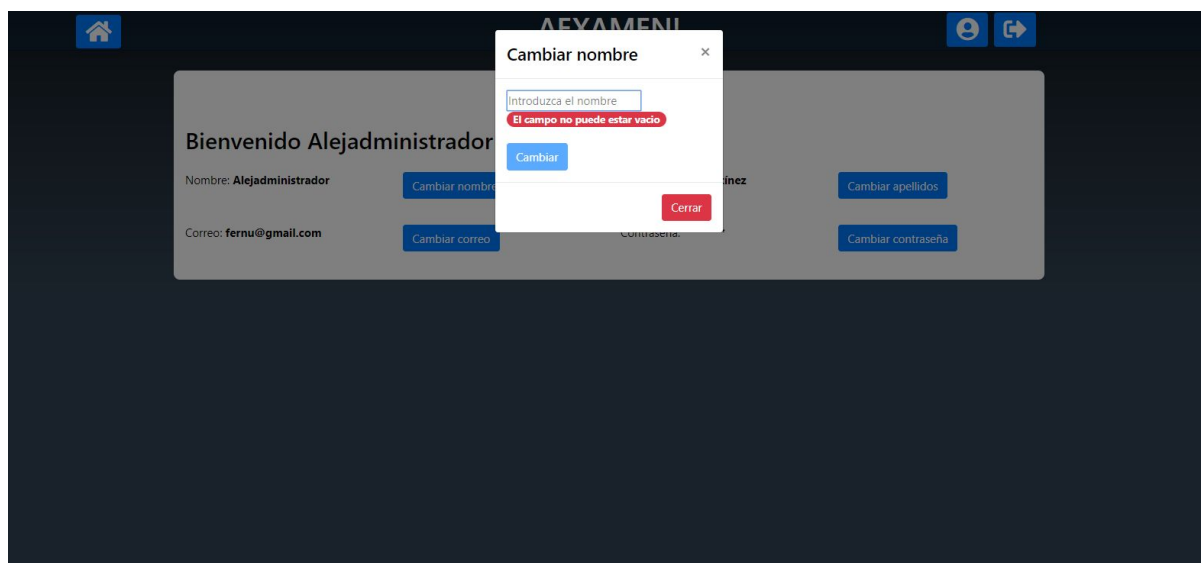


Figura 20. Perfil propio, cambio de nombre

```
<!-- Modal de Nombre -->
<div class="modal" id="modal_cambiarNombre">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">

      <!-- Modal Header de Nombre -->
      <div class="modal-header">
        <h4 class="modal-title">Cambiar nombre</h4>
        <button type="button" class="close" data-dismiss="modal">&times;</button>
      </div>

      <!-- Modal body de Nombre -->
      <div class="modal-body">
        <form action="/perfiles/cambiodatos" class="form-container" method="post" id="form_cambiarNombre">

          <input type="text" placeholder="Introduzca el nombre" name="nombre" id="nombre">

          <button type="submit" class="btn" id="boton_cambiarNombre" name="boton_cambiarNombre">Cambiar</button>
        </form>
      </div>

      <!-- Modal footer de Nombre -->
      <div class="modal-footer">
        <button type="button" class="btn btn-danger" data-dismiss="modal">Cerrar</button>
      </div>

    </div>
  </div>
</div>
```

Figura 21. Modal de cambio de nombre en perfil propio (View\Perfiles\index.ctp)

### 7.3.9 Asignaturas de profesor

Esta vista (Figura 22) carga en el controlador mediante una llamada (Figura 23) al modelo (el cual carga la información de la base de datos) todas las asignaturas que imparte y/o coordina un profesor (indicando en cada una si es coordinador o no). Mediante un click en una de ellas accederá a la vista de una asignatura donde podrá acceder a información

de esta. Además se puede buscar en la barra de búsqueda superior la cual filtra dinámicamente entre las asignaturas presentes.



Siglas	Nombre asignatura	Coordinador
IS	Ingeniería del software	Si
FP	Fundamentos de la programación	No
DR	Derecho Romano	Si

Figura 22. Asignaturas de un profesor

```
public function cargaAsignaturas($idProfesor){
    $sql = "SELECT prof_asig_coord.coordinador AS coordinador, profesores.nombre AS nombre_profesor,
    asignaturas.nombre AS nombre_asignatura, asignaturas.siglas AS siglas_asignatura,
    asignaturas.id AS id_asignatura
    FROM ((prof_asig_coord INNER JOIN profesores ON prof_asig_coord.id_profesor = profesores.id)
    INNER JOIN asignaturas ON prof_asig_coord.id_asignatura = asignaturas.id)
    WHERE id_profesor=".$idProfesor;
    $asignaturas=$this->query($sql);
    /*En caso de que no haya ninguna asignatura, lo señalamos en
    la variable session que controla ese error*/
    if(count($asignaturas)==0){
        $_SESSION['error_ningunaAsignatura']=true;
    }
    return $asignaturas;
}
```

Figura 23. Función del modelo Asignatura que carga de base de datos la información de las asignaturas correspondientes a un profesor (Model\Asignatura.php)

### 7.3.10 Una asignatura

Esta vista muestra enlaces a las preguntas y exámenes (Figura 24) de una asignatura en concreto y además, si el usuario es coordinador de la asignatura (lo cual se comprueba mediante una variable de sesión en la figura 25), también a gestionar los profesores y los parámetros por defecto de dicha asignatura. Se usa el mismo tipo de enlaces que en la página principal del sistema.



Figura 24. Asignatura Ingeniería del Software vista por un coordinador de esta.

```
<?php
//Si es coordinador mostramos las demás opciones
if($esCoordinador){
    ?>
    <div class="col-xl-3 col-lg-4 col-md-6 col-sm-3 portfolio-item filter-app">
        <h3>PROFESORES</h3>
        <a <?php echo 'href="/profesores/profesores_asignatura?idAsig='.$_GET['id'].'&nombreAsig='.$nombreAsig.'"' ?>>
            
            <h3>PROFESORES</h3>
            <div class="details">
                <h4>PROFESORES</h4>
                <span>Gestionar profesores</span>
            </div>
        </a>
    </div>

    <div class="col-xl-3 col-lg-4 col-md-6 col-sm-3 portfolio-item filter-app">
        <h3>PARÁMETROS</h3>
        <a <?php echo 'href="/profesores/definir_parametros?idAsig='.$_GET['id'].'"' ?>>
            
            <h3>PARÁMETROS</h3>
            <div class="details">
                <h4>PARÁMETROS DE EXAMEN</h4>
                <span>Definir parámetros de exámenes</span>
            </div>
        </a>
    </div>
<?php
}
```

Figura 25. Parte de la vista que, de tratarse de un coordinador, muestra los enlaces a Profesores y a Parámetros de la asignatura (View\Asignaturas\una\_asignatura.ctp).

### 7.3.11 Gestionar profesores de una asignatura

Esta vista sólo es accesible para coordinadores de una asignatura (Figura 26). Esto se comprueba inmediatamente en el controlador antes de cargar la vista. En esta vista se pueden añadir un profesor a la asignatura mediante la selección de este en el modal que aparece al pulsar el botón de “+”. Los profesores que imparten la asignatura se cargan de base de datos a través del modelo y se muestran en forma de tabla pasando sus datos del controlador a la vista (Figura 27). También se puede eliminar cada profesor de la asignatura pulsando sobre el icono de papelera roja y confirmando la acción (de modo que no se elimine nada de forma accidental).

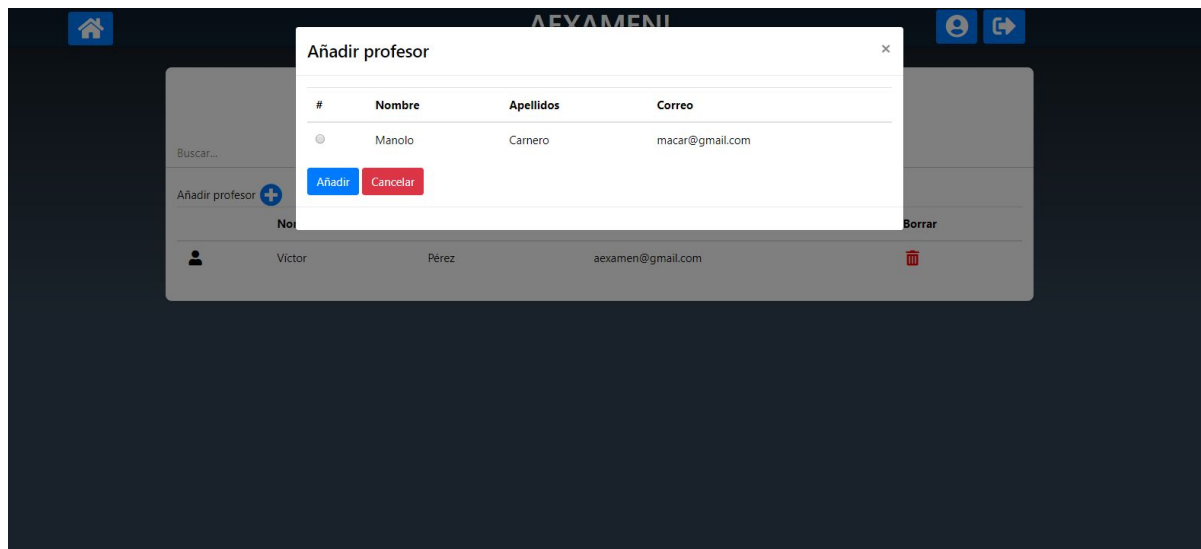


Figura 26. Añadir un profesor a una asignatura.

```
$.ajax({
  type      : 'POST',
  url       : '/profesores/funcionesAjaxProfesores',
  data      : 'funcion=' + funcion + '&idAsig=' + idAsig + '&idProfesores=' + idProfesores,
  success: function(respuesta){
    console.log(respuesta);
    if(respuesta){
      console.log(respuesta);
      $('#boton_anadir').attr('disabled', false);
      $('#tableAniadirProfesor').children('tr,td').remove();
      $('#infoTodosProfAdd').hide();
      if(respuesta.length > 0){
        var pinta = false;
        for (i = 0; i < respuesta.length; i++) {
          $('#tableAniadirProfesor').append('<tr><td><input type="radio" name="profesor" value="'+respuesta[i]["id"]+'"></td><td>'+respuesta[i]["nombre"]+'</td><td>'+respuesta[i]["apellidos"]+'</td><td>'+respuesta[i]["email"]+'</td></tr>');
        }
      }
      else{
        $('#infoTodosProfAdd').show();
      }
      $('#modalAniadirProfesor').modal('show');
    }
    else{
      console.log("falla");
    }
  },
  dataType: "json"
})
```

Figura 27. Llamada AJAX que obtiene todos los profesores que no están asignados a la asignatura y los pinta en la tabla para su elección (webroot\js\profesoresDeUnaAsig.js).

### 7.3.12 Gestionar asignaturas y coordinadores

En esta vista (sólo accesible para administradores) se cargan todas las asignaturas del sistema mediante la función `cargaTodasAsignaturas` del modelo `Asignatura` de modo que se muestran en la vista las siglas, el nombre, el número de profesores que la imparten y los nombres de los coordinadores de esta. Además se pueden quitar o añadir coordinadores a cada asignatura mediante el botón “+”. Cuando se pulsa este botón, se carga dinámicamente mediante una llamada AJAX (Figura 29) una lista de profesores pertenecientes a la asignatura correspondiente cada uno con un checkbox para poder seleccionar o deseleccionar varios al mismo tiempo y guardar los nuevos coordinadores de la asignatura (Figura 28).

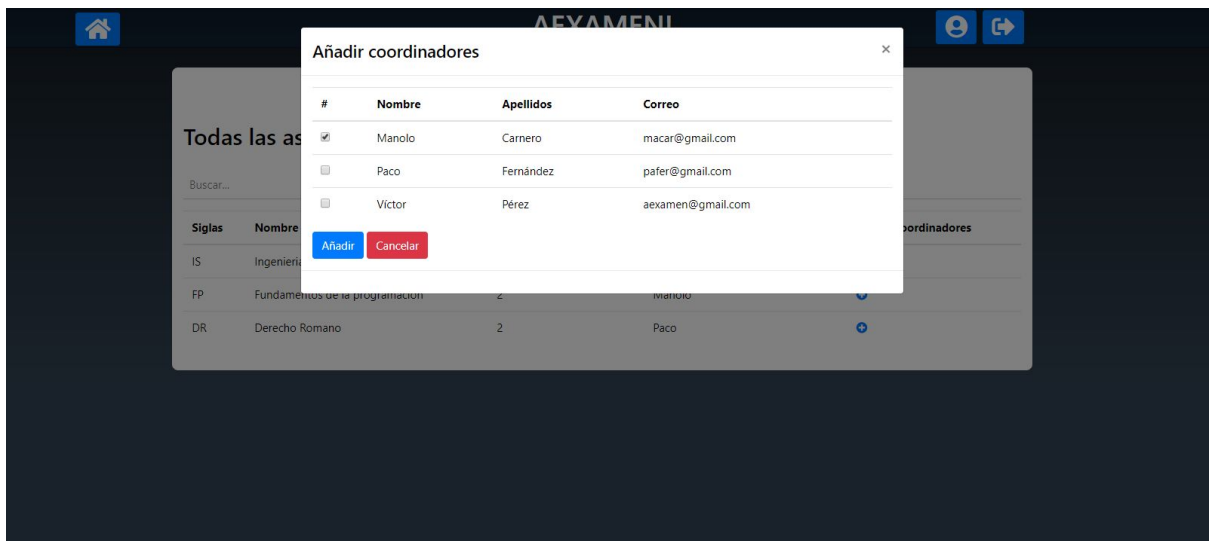


Figura 28. Añadir coordinadores a una asignatura.

```
$.ajax({
  type      : 'POST',
  url       : '/asignaturas/ajaxasignaturas',
  data      : 'funcion=' + funcion + '&idAsig=' + idAsig,
  success: function(respuesta){
    if(respuesta){
      $("#formAsigCoord").attr("idAsig", idAsig);
      console.log(respuesta);
      $("#boton_anadir").attr('disabled', false);
      $("#tableCoordinadores").children('tr,td').remove();
      let hayProf = false;
      for (var i = 0; i < respuesta['profSiCoord'].length; i++) {
        hayProf = true;
        console.log(respuesta['profSiCoord'][i]);
        $("#tableCoordinadores").append('<tr><td><input type="checkbox" checked name="profesor" value="'+respuesta['profSiCoord'][i]['id']+'"'></td><td>'+respuesta['profSiCoord'][i]['nombre']+'</td><td>'+respuesta['profSiCoord'][i]['apellidos']+'</td><td>'+respuesta['profSiCoord'][i]['email']+'</td></tr>');
      }
      for (var i = 0; i < respuesta['profNoCoord'].length; i++) {
        hayProf = true;
        console.log(respuesta['profNoCoord'][i]);
        $("#tableCoordinadores").append('<tr><td><input type="checkbox" name="profesor" value="'+respuesta['profNoCoord'][i]['id']+'"'></td><td>'+respuesta['profNoCoord'][i]['nombre']+'</td><td>'+respuesta['profNoCoord'][i]['apellidos']+'</td><td>'+respuesta['profNoCoord'][i]['email']+'</td></tr>');
      }
      if(!hayProf){
        $("#message").append('<div class="alert alert-warning"><a href="#" class="close" data-dismiss="alert" aria-label="close">&times;</a> No hay ningún profesor en el sistema</div>');
      }
    }
  }
});
else{
  console.log("falla");
}
```



Figura 29. Parte de la llamada AJAX que recibe la lista de profesores de una asignatura y los ordena para poder seleccionarlos para coordinar dicha asignatura (webroot\js\gestionarAsigAdmin.js)

### 7.3.13 Gestionar profesores del sistema

Esta vista (accesible sólo para administradores) muestra una lista de todos los profesores dados de alta en el sistema obtenida mediante una llamada al modelo Profesor desde el controlador ProfesoresController. También es posible cambiar los datos de un profesor haciendo click sobre el icono de lápiz. Se podrá eliminar a un profesor de la plataforma pulsando sobre la papelera roja confirmando el modal. Para elegir qué asignaturas coordina un profesor en particular se puede pulsar sobre Asignaturas coordinadas y se cargará un modal con una tabla de asignaturas que coordina y que no coordina (Figura 30), con checkboxes de modo que se podrá seleccionar las asignaturas deseadas haciendo click en Añadir mediante AJAX (Figura 31). Esto es posible siempre y cuando la selección de asignaturas no deje una asignatura sin coordinador, en cuyo caso surgirá un error y no se realizarán los cambios de coordinadores.

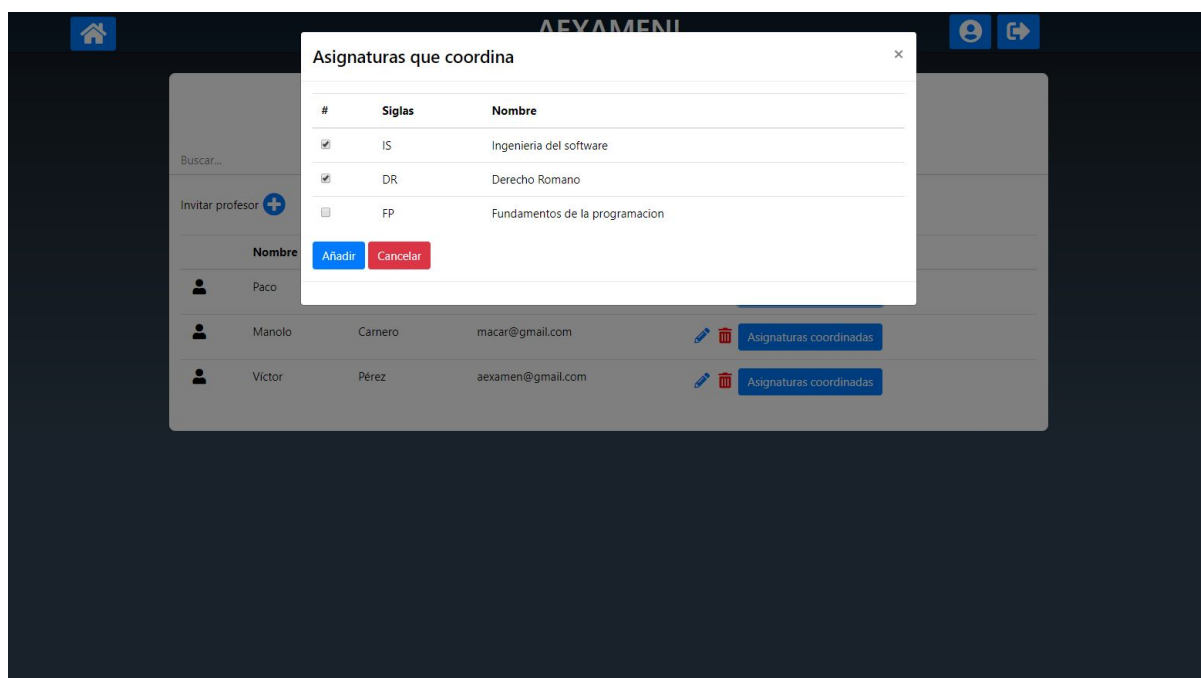


Figura 30. Asignaturas que coordina un profesor y formulario para cambiarlas.

```

$.ajax({
  type      : 'POST',
  url       : '/profesores/funcionesajaxprofesores',
  data      : 'funcion=' + funcion + '&idProfesor=' + idProfesor,
  success: function(respuesta){
    if(respuesta){
      console.log(idProfesor);
      $("#formAsig").attr("idProfesor", idProfesor);
      console.log(respuesta);
      console.log("llega");
      $('#tableAsignaturas').children('tr,td').remove();

      for (var i = 0; i < respuesta['asigSiCoord'].length; i++) {
        console.log(respuesta['asigSiCoord'][i]);
        $('#tableAsignaturas').append('<tr><td><input type="checkbox" checked name="asignatura" value="'+
          respuesta['asigSiCoord'][i]['id']+" class="asigCheckbox"></td><td>'+respuesta['asigSiCoord'][i]['siglas']+'
          '</td><td>'+respuesta['asigSiCoord'][i]['nombre']);
      }
      for (var i = 0; i < respuesta['asigNoCoord'].length; i++) {
        console.log(respuesta['asigNoCoord'][i]);
        $('#tableAsignaturas').append('<tr><td><input type="checkbox" name="asignatura" value="'+
          respuesta['asigNoCoord'][i]['id']+" class="asigCheckbox"></td><td>'+respuesta['asigNoCoord'][i]['siglas']+'
          '</td><td>'+respuesta['asigNoCoord'][i]['nombre']);
      }
    }
    else{
      console.log("falla");
    }
  },
  dataType: "json"
})

```

Figura 31. Petición AJAX que obtiene las asignaturas correspondientes a un profesor y las imprime en el modal de la Figura 30 empezando por las que coordina seguidas de las que no (webroot\js\profesoresAdmin.js)

### 7.3.14 Login

Al acceder al login, se comprueba en el controlador si el usuario ya está logueado, en tal caso se redirige a la página principal. En caso de que no sea así se permite al usuario rellenar los campos y cuando este envía el formulario, se capturan estos en el controlador y este llama al modelo para que haga el login. En el modelo se comprueba con la base de datos si el usuario y la contraseña son correctos y si se trata de un usuario profesor o administrador. Tras esto devuelve al controlador si se ha realizado con éxito o no el login. El controlador si se ha realizado el login de forma correcta redirecciona a PaginasPrincipales, y si es de forma incorrecta vuelve a la vista de Login, pasándole a esta un mensaje de error para que lo muestre.



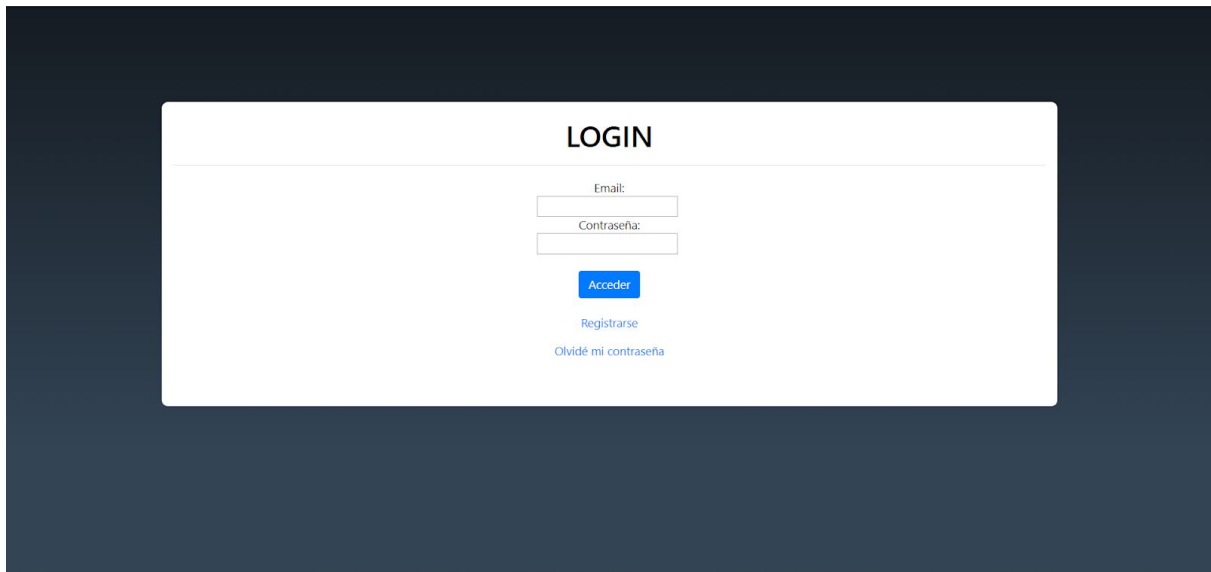


Figura 32. Pantalla del login de la aplicación.

```
//Verificamos la clave con esta funcion ya que en la BBDD esta encriptada, en caso de que se verifique,
// declaramos e inicializamos todas las variables de session de usuario.
$datos = ($encontrado) ? $filaEncontradaUser : $filaEncontradaAdmin;
if(password_verify($clave, $datos['clave'])){
    $_SESSION['logeado']=true;
    $_SESSION["email"] = $email;
    $_SESSION["nombre"]=$datos['nombre'];
    $_SESSION['apellidos']=$datos['apellidos'];
    $_SESSION['id']=$datos['id'];
    $_SESSION['administrador'] = $encontradoAdmin;
    date_default_timezone_set("Europe/Madrid");
    //Registramos este login en el log
    $log = '['.date("d/m/Y - H:i:s").'] : '. "USER --> id ".$_SESSION['id'].' - '.$_SESSION['apellidos'].' , '.$_SESSION['nombre'].
        " | ACTION --> Inicio de sesión ". ' de '.$_SESSION['email'].PHP_EOL.
        "-----".PHP_EOL;
    file_put_contents('log/log_AExamen.log', utf8_decode($log), FILE_APPEND);


    return true;
}
```

Figura 33. Código del modelo de Login. En él se comprueba si la contraseña introducida es la misma a la contraseña hasheada que se sitúa en la base de datos. En caso de que sea así se cargan todos los datos en la variable session.

### 7.3.15 Restablecer contraseña

El usuario introduce su mail en el formulario y lo envía. El correo llega al controlador que llama al modelo pasándole por parámetro el mail que ha introducido el usuario. En el modelo se comprueba si este correo existe en la base de datos, en caso de que no existe se lo notifica al controlador y este a la vista, que muestra el error. En caso de que si exista, se le manda un correo al email proporcionado y se le notifica al controlador y este se lo notifica a la vista que muestra el mensaje correspondiente.

El email se manda desde un correo el cual está guardado en un json del proyecto, junto con su contraseña. Para mandar un correo se usa el objeto PHPMailer, al cual se le pasan una serie de parámetros.



**RESTABLECER CONTRASEÑA**

Introduce tu correo electrónico para poder restablecer la contraseña mediante el correo que te enviaremos:

Email \*:  
email@example.com

[Enviar](#)

[Iniciar sesión](#)  
[Registrarse](#)

Figura 34. Pantalla de restablecer contraseña.

```

public function smtpmailerRaw($to, $from, $fromName, $subject, $body, $googleUser, $googlePassword) {
    global $error;
    $mail = new PHPMailer(); // creamos el objeto
    $mail->IsSMTP(); // activa SMTP
    $mail->SMTPDebug = 0; // debugeo: 1 = errores y mensajes, 2 = sólo mensajes
    $mail->SMTPAuth = true; // requerir autenticación
    $mail->SMTPSecure = 'ssl'; // transferencia segura activada OBLIGATORIO para GMail
    $mail->Host = 'smtp.gmail.com';
    $mail->Port = 465;
    $mail->Username = $googleUser;
    $mail->Password = $googlePassword;
    $mail->SetFrom($from, $fromName);
    $mail->Subject = $subject;
    $mail->AddAddress($to);
    $mail->Body = $body;
    $mail->CharSet = 'ISO-8859';
    $mail->isHTML(true);
    if(!$mail->Send()) {
        $error = 'Mail error: '.$mail->ErrorInfo;
        return false;
    } else {
        $error = "\nMessage sent to ".$to."!";
        return true;
    }
}

```

Figura 35. Código a través del cual se usa el objeto PHPMailer para enviar un correo al email proporcionado.

### 7.3.16 Registrarse

Antes de que se muestre la vista, el controlador comprueba si el usuario que está intentando acceder a esta página está o no está logueado, en caso de que esté logueado lo redirige a la paginaPrincipal. El usuario rellena en la vista un formulario con una serie de campos necesarios para poder registrarse. Una vez que los ha rellenado los envía. El controlador recibe los campos introducidos por el usuario. En el controlador se comprueba si se ha realizado una petición POST, es decir, si se ha enviado el formulario, en caso de que sea así carga los valores y llama a una función del modelo pasándole los valores por parámetro.

En el modelo se comprueba si este email ya está dado de alta en la aplicación a través de la base de datos. En caso de que no esté dado de alta, inserta la petición en la tabla 'peticiones\_registro' y envía un correo al usuario explicándole que su petición ya está en curso.

Figura 36. Pantalla de registrarse.

```
public function registrarse($email, $nombre, $apellidos, $clave, $texto){
    $sql = "SELECT * FROM profesores WHERE email='".$email."'";
    $consulta=$this->query($sql);

    if(count($consulta) > 0) {
        $_SESSION['error_usuario_existente']=true;
        return false;
    } else {
        $_SESSION['error_envio_mail'] = false;
        $credentialsStr = file_get_contents('json/credentials.json');
        $credentials = json_decode($credentialsStr, true);
        if (!$this->smtpmailer($email, $credentials['webMail']['mail'], 'AExamen Web', 'Solicitud de registro AExamen', 'registroAexamen-mail.html', $credentials['webMail']['mail'],
            $credentials['webMail']['password'])) {
            $_SESSION['error_envio_mail'] = true;
        }

        date_default_timezone_set('Europe/Berlin');
        $date = date('Y-m-d H:i:s', time());
        $claveHash = password_hash($clave, PASSWORD_BCRYPT);
        $sql = "INSERT INTO 'peticiones_registro'('id', 'nombre', 'apellidos', 'email', 'fecha', 'texto', 'clave') VALUES ('', '".$nombre."', '".$apellidos."', '".$email."', '".$date."',
            '".$texto."', '".$claveHash."')";
        $consulta=$this->query($sql);

        return true;
    }
}
```

Figura 37. Código del modelo de registrarse. Se observa cómo se llama a la función la cual manda el email y cómo se inserta en la base de datos la petición.

### 7.3.17 Exámenes

Antes de acceder a la vista, el controlador comprueba si el usuario está logueado, en caso de no ser así lo redirige a la ruta index ('/'). Tras esto el controlador llama al modelo para cargar todos los datos necesarios para mostrarlos en la vista. En el modelo se realizan las diversas peticiones a la base de datos y devuelve los datos obtenidos al controlador. Una vez que el controlador obtiene todos los datos del modelo, se los proporciona a la vista y esta los muestra.

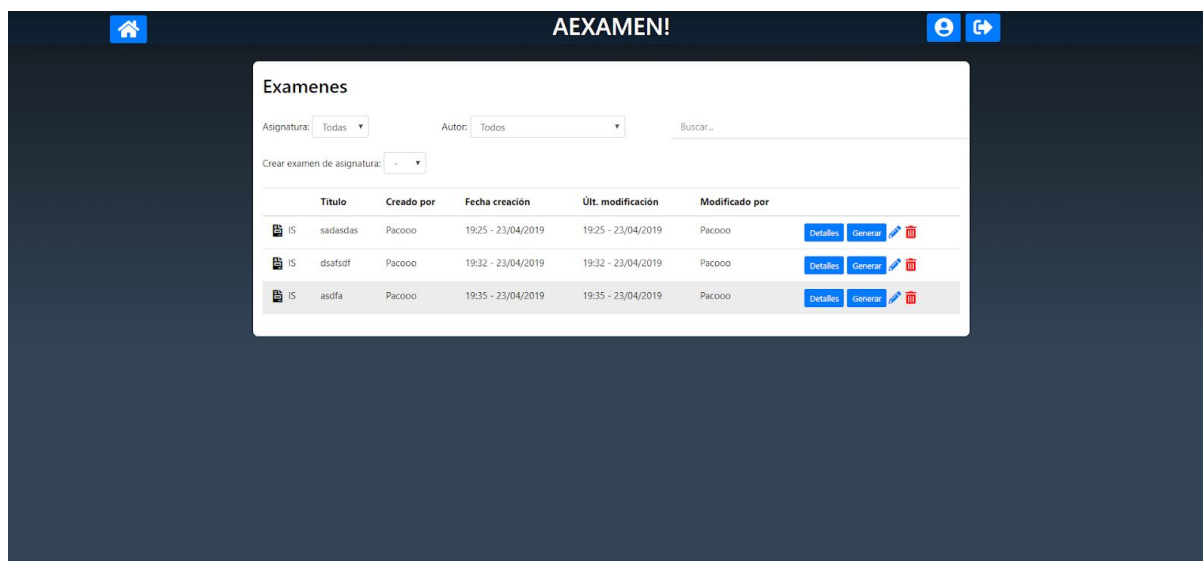


Figura 38. Pantalla de exámenes de la aplicación.

```

class ExamenesController extends ApplicationController {
    public function index(){
        /*Iniciamos la sesion, pero antes hacemos una comprobacion para evitar errores*/
        if (session_status() == PHP_SESSION_NONE) {
            session_start();
        }
        //Si existe $_SESSION['logeado'] volcamos su valor a la variable, si no existe volcamos false. Si vale true es que estamos logeado.
        $logeado = isset($_SESSION['logeado'])? $_SESSION['logeado']: false;
        /*En caso de no este logeado redirigimos al login, en caso contrario le damos la bienvenida*/
        if (!$logeado) {
            return $this->redirect('/');
        }

        //Si es administrador mostramos en el filtro todas las asignaturas
        //en caso de que sea un profesor solo mostramos las asignaturas que
        //tenga asignadas
        $this->loadModel('Examen');
        if ($_SESSION['administrador']) {
            $siglas = $this->Examen->selectAllSiglasAsignaturas();
        } else {
            $siglas = $this->Examen->selectAllSiglasAsignaturasProfesor($_SESSION['id']);
        }

        $autores = $this->Examen->selectAllMailsProfesoresSiglas($_GET['asignatura']);

        if ($_GET['asignatura'] == "todas" && $_GET['autor'] == "todos") {
            $examenes = $this->Examen->selectAllExamenesCompleto();
        } else {
            $examenes = $this->Examen->selectAllExamenesFiltrado($_GET['asignatura'], $_GET['autor']);
        }
        $this->set('siglas', $siglas);
        $this->set('autores', $autores);
        $this->set('examenes', $examenes);
    }
}

```

Figura 39. Código del controlador de registrarse. Se observa cómo se llama a las diferentes funciones del modelo para cargar los datos de la base de datos.

### 7.3.18 Detalles examen.

Antes de acceder a la vista, el controlador comprueba si el usuario está logueado, en caso de no ser así lo redirige a la ruta index ('/'). Tras esto el controlador llama al modelo para cargar todos los datos necesarios para mostrarlos en la vista, tales como datos del examen, el nombre del autor del examen, la fecha de creación del examen formateada, las preguntas del examen y el historial de modificaciones. En el modelo se realizan las diversas

peticiones a la base de datos y devuelve los datos obtenidos al controlador. Cuando el controlador obtiene toda la información, se la proporciona a la vista.

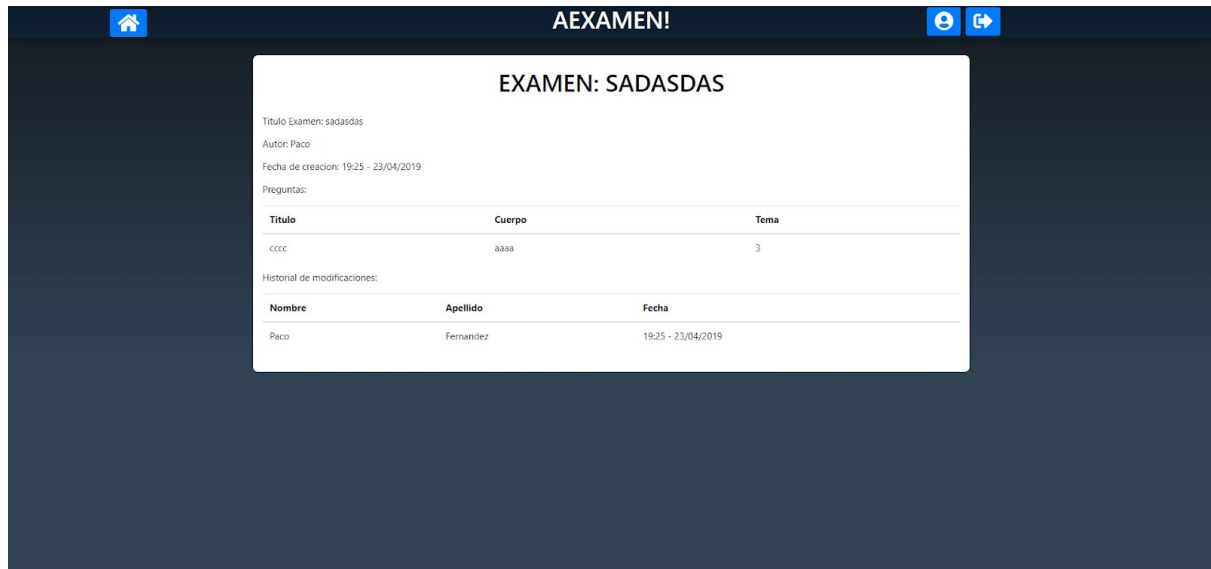


Figura 40. Pantalla de examen.

```

/*
 * Función para formatear una fecha de formato estándar string a diferentes formatos según cuanto
 * tiempo haya pasado hasta la actualidad
 */
public function formateoDateTime ($fecha) {
    date_default_timezone_set("Europe/Madrid");

    $time = strtotime($fecha);

    $diff = (time() - $time)*1000; // La diferencia en milisegundos

    if ($diff < 1000) { // menos de un segundo
        return 'ahora mismo';
    }

    $sec = floor($diff / 1000); // diferencia en segundos

    if ($sec < 60) { // menos de un minuto
        return 'hace '.$sec.' seg.';
    }

    $min = floor($diff / 60000); // diferencia en minutos

    if ($min < 60) { // menos de una hora
        return 'hace '.$min.' min.';
    }

    //si el día coincide
    if ((date("d")==date('d',$time)) && (date("m")==date('m',$time)) && (date("Y")==date('Y',$time))) {
        return date('H:i',$time);
    }

    $newformat = date('H:i - d/m/Y',$time);

    return $newformat;
}

```

Figura 41. Código del modelo Examen. Función que devuelve la diferencia entre una fecha y la actual

### 7.3.19 Borrar examen

Cuando en la vista de exámenes se pulse sobre el icono de la papelera, se muestra un pop-up que es lanzado desde el fichero JavaScript 'examen.js'. Este listener se realiza mediante JQuery. El pop-up preguntará si se quiere borrar el examen y cuando se pulse a 'Sí' se le mostrará otra ventana de confirmación para evitar posibles errores, todas estas comprobaciones se realizan desde el JavaScript. Una vez que se haya confirmado, el JavaScript hará una llamada AJAX al controlador. Este comprobará los parámetros y según los que sean llamará a la función correspondiente del modelo. En el modelo se borrará el examen de la base de datos con la información proporcionada.

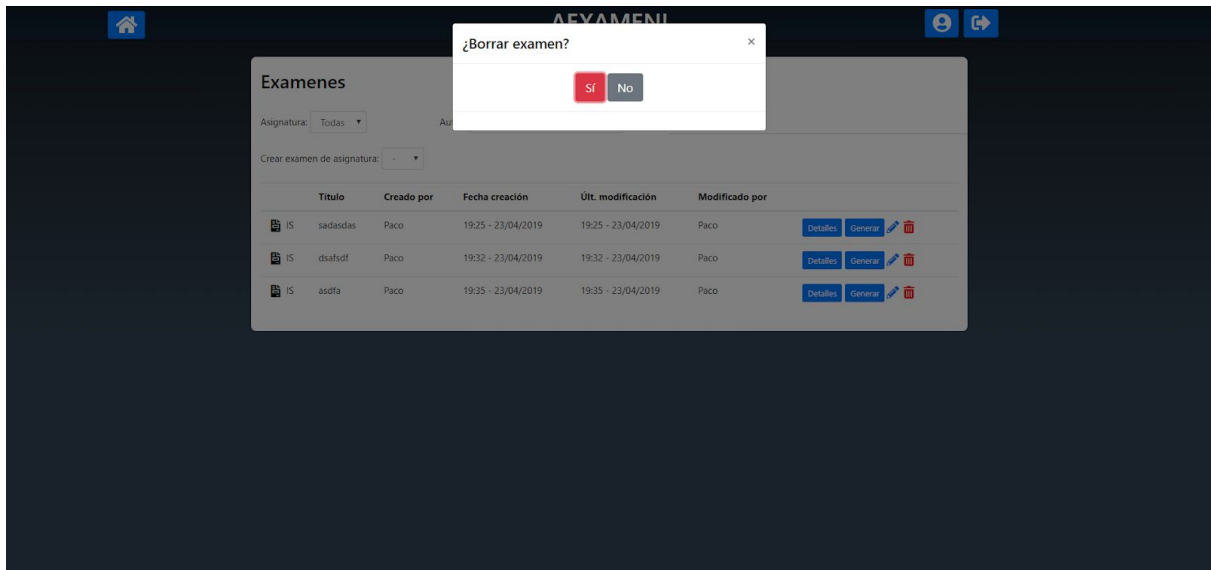


Figura 42. Pantalla de borrar examen.

```

$('#opciones a').click(function() {
    var id = $(this).attr("id");
    if(id == "boton_modalEditar") {
        $("#boton_editar").attr("id_examen",$(this).attr("idExamen"));
        $("#modal_editarExamen").modal('show');
    }
    else if(id == "boton_modalBorrar"){
        $("#boton_borrar").attr("id_examen",$(this).attr("idExamen"));
        $("#modal_borrarExamen").modal('show');
    }
}

```

Figura 43. Código del JavaScript del examen. En él se puede observar los eventos que se realizan para mostrar los modales.

### 7.3.20 Preguntas

El controlador llama al modelo para cargar todos los datos necesarios para mostrarlos en la vista. En el modelo se realizan las diversas peticiones a la base de datos y devuelve los datos obtenidos al controlador. Una vez que el controlador obtiene todos los datos del modelo, se los proporciona a la vista y esta los muestra.





Figura 44. Pantalla de pregunta.

```
function cargaPreguntas($idAsignatura, $emailAutor){
    $sql = "SELECT asignaturas.siglas AS siglasAsignatura, profesores.nombre AS autor, preguntas.titulo AS titulo, preguntas.cuerpo AS cuerpo,
preguntas.tema AS tema, preguntas.fecha_creacion AS fecha_creacion, preguntas.fecha_modificado AS fecha_modificado, preguntas.id AS id_preguntas
FROM ((preguntas INNER JOIN asignaturas ON asignaturas.id = ".$idAsignatura.") INNER JOIN profesores ON preguntas.creador=profesores.id)
WHERE preguntas.asignatura=".$idAsignatura;

    if ($emailAutor != "todos") {
        $sql = $sql." AND profesores.email='".$emailAutor."'";
    }
    $consulta=$this->query($sql);
    $resultado = [];

    if(count($consulta) > 0){
        $count = 0;
        while (count($consulta) > $count){
            foreach ($consulta[$count]["preguntas"] as $key => $value) {
                $resultado[$count][$key] = $consulta[$count]["preguntas"][$key];
            }
            $resultado[$count]["autor"] = $consulta[$count]["profesores"]["autor"];
            $resultado[$count]["fecha_creado_raw"] = $this->formatoDateTime($consulta[$count]["preguntas"]["fecha_creacion"]);
            $resultado[$count]["fecha_modificado_raw"] = $this->formatoDateTime($consulta[$count]["preguntas"]["fecha_modificado"]);
            $count++;
        }
    } else {
        $resultado = null;
    }
    return $resultado;
}
```

Figura 45. Código del modelo de pregunta. Carga todas las preguntas de una asignatura y de un autor

### 7.3.21 Detalles pregunta

El controlador llama al modelo para cargar todos los datos necesarios para mostrarlos en la vista, tales el historial de cambios y datos de la pregunta. En el modelo se realizan las diversas peticiones a la base de datos y devuelve los datos obtenidos al controlador. Cuando el controlador obtiene toda la información, se la proporciona a la vista.



Figura 46. Pantalla de detalle de pregunta.

```
function cargaHistorialPregunta($idPregunta){
    $sql = "SELECT `id`, `idPregunta`, `idModificador`, `fecha_modificacion` FROM `preguntas_historial` WHERE `idPregunta`=",$idPregunta;
    $consulta=$this->query($sql);
    $resultado = [];
    if(count($consulta) > 0){
        $count = 0;
        while (count($consulta) > $count){
            foreach ($consulta[$count]["preguntas_historial"] as $key => $value) {
                $resultado[$count][$key] = $consulta[$count]["preguntas_historial"][$key];
            }
            $nombreApellidosAutor= $this->cargaNombreApellidosAutor($consulta[$count]["preguntas_historial"]["idModificador"]);
            $resultado[$count]["nombreAutor"] = $nombreApellidosAutor['profesores']['nombre'];
            $resultado[$count]["apellidosAutor"] = $nombreApellidosAutor['profesores']['apellidos'];
            $resultado[$count]["fecha_modificado_raw"] = $this->formateoDateTime($consulta[$count]["preguntas_historial"]["fecha_modificacion"]);
            $count++;
        }
    } else {
        $resultado = null;
    }
    return $resultado;
}
```

Figura 47. Código de modelo de pregunta el cual retorna el historial de cambios de la pregunta.

### 7.3.22 Editar pregunta

Cuando en la vista de exámenes se pulse sobre el icono del lápiz, se muestra un pop-up que es lanzado desde su fichero JavaScript. Este listener se realiza mediante JQuery. El pop-up contendrá un formulario con tres campos: título, cuerpo y tema. Una vez que se haya rellenado el formulario y se envíe, el JavaScript hará una llamada AJAX al controlador. Este comprobará los parámetros y según los que sean llamará a la función correspondiente del modelo. En el modelo editará la pregunta de la base de datos con la información proporcionada.

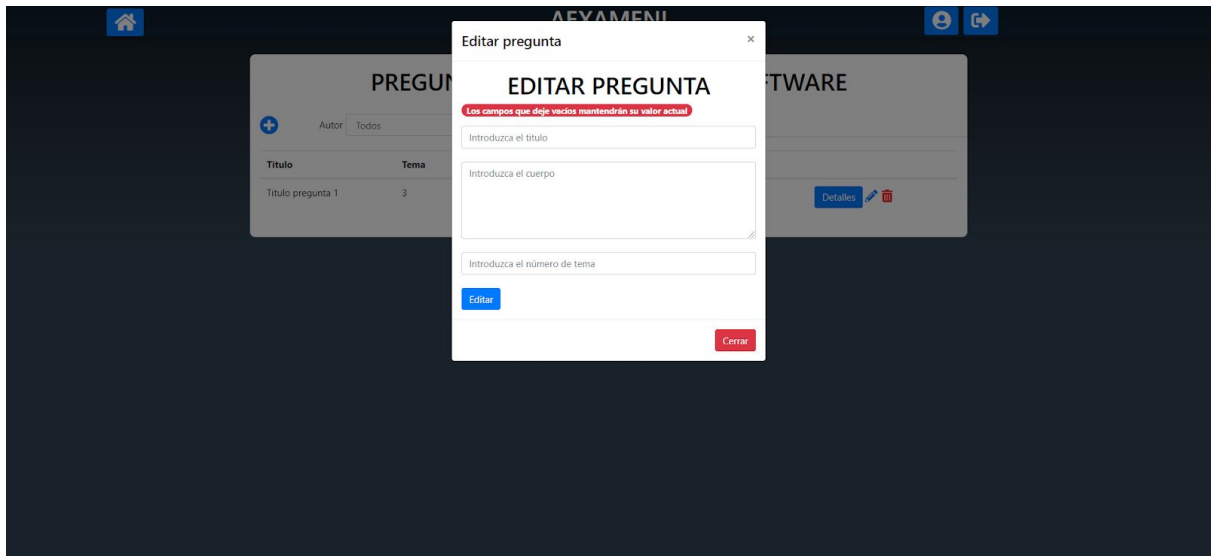


Figura 48. Pantalla de editar pregunta.

```
//Submit del formulario de modificar
$('#form_mod').submit(function(event) {
    var funcion = "editarPregunta";
    var form_data = $(this).serialize();
    $.ajax({
        type      : 'POST',
        url       : '/preguntas/funcionesajaxpreguntas',
        data      : form_data + '&funcion=' + funcion + '&id_pregunta=' + $("#boton_editar").attr("id_pregunta"),
        success: function(respuesta) {
            if(respuesta){
                location.reload();
            }
            else{
                location.reload();
            }
        }
    })
    event.preventDefault();
});
```

Figura 48. Código AJAX de preguntas.js que se ejecuta cuando se hace submit al formulario de editar pregunta.

### 7.3.23 Borrar pregunta

Cuando en la vista de exámenes se pulse sobre el icono de la papelera, se muestra un pop-up que es lanzado desde su fichero JavaScript. Este listener se realiza mediante JQuery. El pop-up preguntará si se quiere borrar la pregunta y cuando se pulse a 'Si' se le mostrará otra ventana de confirmación para evitar posibles errores, todas estas comprobaciones se realizan desde el JavaScript. Una vez que se haya confirmado, el JavaScript hará una llamada AJAX al controlador. Este comprobará los parámetros y según los que sean llamará a la función correspondiente del modelo. En el modelo se borrará el examen de la base de datos con la información proporcionada.

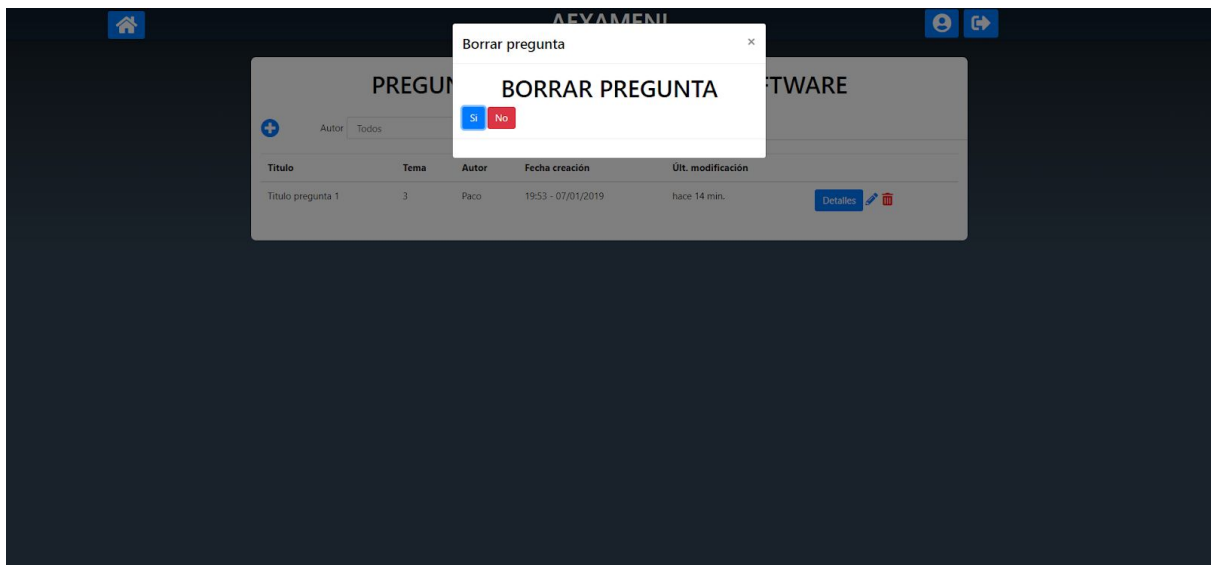


Figura 50. Pantalla de borrar pregunta.

```
public function funcionesAjaxPreguntas(){
    $_SESSION['error_no_poder_borrar'] = false;
    $this->layout= 'ajax';
    $this->render(false);
    $titulo = isset($_POST['titulo'])? $_POST['titulo']: null;
    $cuerpo = isset($_POST['cuerpo'])? $_POST['cuerpo']: null;
    $tema = isset($_POST['tema'])? $_POST['tema']: null;
    $funcion = isset($_POST['funcion'])? $_POST['funcion']: null;
    $idPregunta = isset($_POST['id_pregunta'])? $_POST['id_pregunta']: null;
    if($funcion == "aniadirPregunta"){
        $aniadirPregunta = $this->Pregunta->aniadirPregunta($titulo,$cuerpo,$tema);
        echo $aniadirPregunta;
    }
    else if($funcion == "borrarPregunta"){
        $borrarPregunta = $this->Pregunta->borrarPregunta($idPregunta);
        echo $borrarPregunta;
    }
    else if($funcion == "editarPregunta"){
        $editarPregunta = $this->Pregunta->editarPregunta($titulo,$cuerpo,$tema,$idPregunta);
        echo $editarPregunta;
    }
}
```

Figura 51. Código del controlador de Pregunta. Función a la cual llaman los diversos Ajax. En consecuencia de que le pasen por POST, llamará a una u otra función del modelo.

## 8 Evaluación

Para evaluar la aplicación se ha realizado una encuesta con Google Forms. Se ha distribuido a través de redes sociales, por lo que la han realizado todo tipo de perfiles. Dentro del formulario se presentó la aplicación describiendo en qué consiste y dando unos consejos antes de llevarla a cabo.

La encuesta consiste en realizar una serie de acciones propuestas para posteriormente valorar la experiencia sentida por el usuario al llevarlas a cabo.

### 8.1 Estructura de la encuesta

La encuesta consta de las siguientes partes:

- Introducción: primero se pone al usuario encuestado en situación, se le recomienda realizar el proceso en un ordenador y se le explica que hará el rol de un profesor de universidad. Se pide la edad, el correo electrónico, el sexo y se le pregunta si actualmente trabaja en la enseñanza. (Figura 52)

## Encuesta de satisfacción AExamen



A continuación aparecerán una serie de acciones que deberás realizar en nuestra página web para luego contestar a las preguntas correspondientes.

Es **IMPORTANTE** USAR un ORDENADOR 🖥️ (ya sea portátil o de sobremesa) y no un smartphone o tablet para las pruebas, puesto que la página web sólo está preparada para pantallas de ordenador.

Has de saber que harás el rol de "profesor/a de universidad" 👨🏫 👩🏫 que utiliza nuestra página web para todo lo relacionado con sus exámenes y asignaturas.

Gracias por participar en la encuesta.

•Obligatorio

**Dirección de correo electrónico \***

Tu dirección de correo electrónico

**Edad \***

Tu respuesta

**Sexo \***

☐ Mujer

☐ Hombre

☐ Prefiero no decirlo

☐ Otro: \_\_\_\_\_

**¿Trabajas en la enseñanza o similares? \***

☐ Sí

☐ No

**SIGUIENTE**

Página 1 de 2

Nunca envíes contraseñas a través de Formularios de Google.

Figura 52. Formulario de evaluación (I)

- Tareas y evaluación: las siguientes 5 cuestiones se tratan de tareas que el usuario debe de realizar dentro de nuestra página (para ello se le proporciona un enlace). Cada vez que considere que ha finalizado, volverá al formulario para puntuar el nivel de satisfacción obtenido realizando dicha tarea. (Figura 53)

## Encuesta de satisfacción AExamen



•Obligatorio

Tareas a realizar en <http://aexamen.epizy.com>

A continuación se enumeran las tareas junto con las cuestiones correspondientes.

1. Accede a <http://aexamen.epizy.com> , inicia sesión con el usuario [aexamen@gmail.com](mailto:aexamen@gmail.com) y contraseña 'test'.

Indica tu nivel de satisfacción con el inicio de sesión \*

	1	2	3	4	5	
Muy bajo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muy alto

2. Cambia el nombre de tu usuario por el de algún familiar o amigo

Indica tu nivel de satisfacción con el cambio de nombre \*

	1	2	3	4	5	
Muy bajo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muy alto

3. Mira las asignaturas disponibles, escoge una de ellas y crea una pregunta

Indica tu nivel de satisfacción creando una pregunta \*

	1	2	3	4	5	
Muy bajo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muy alto

4. Vuelve a las asignaturas disponibles, escoge una de ellas y crea un examen de dicha asignatura incluyendo la pregunta que creaste en el paso 3. Después guarda el examen

Indica tu nivel de satisfacción creando un examen \*

	1	2	3	4	5	
Muy bajo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muy alto

5. Escoge un examen de cualquier asignatura y genera el pdf que llevarás a clase el día del examen.

A veces por problemas de conexión tiene una cierta demora

Indica tu nivel de satisfacción generando el pdf de un examen \*

	1	2	3	4	5	
Muy bajo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muy alto

Figura 53. Formulario de evaluación (II)



- Cuestionario final: por último el usuario debe dar su opinión sobre la facilidad o dificultad presentadas durante la realización de las tareas, evaluar algunos aspectos de la página web y un párrafo opcional de opinión adicional. (Figura 54)

¿Qué es lo que más te ha costado hacer? \*

	Fácil	Medio	Difícil
1. Acceder e iniciar sesión	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2. Cambia el nombre de tu usuario por el de algún familiar o amigo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3. Mira las asignaturas disponibles, escoge una de ellas y crea una pregunta	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4. Vuelve a las asignaturas disponibles, escoge una de ellas y crea un examen de dicha asignatura incluyendo la pregunta que creaste en el paso 3. Después guarda el examen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. Escoge un examen de cualquier asignatura y genera el pdf que llevarás a clase el día del examen.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

¿Qué te ha parecido la página web? \*

1= Muy mala 5= Excelente

	1	2	3	4	5	N/D
Usabilidad	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Colores	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Menús	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

¿Alguna observación relevante? ¡Háznoslo saber!

Tu respuesta

☐ Envíame una copia de mis respuestas.

ATRÁS ENVIAR

Página 2 de 2

Figura 54. Formulario de evaluación (III)



## 8.2 Resultados

A continuación se exponen los resultados obtenidos en la encuesta realizada.

- Edad:

La mayoría de participantes se encuentran entre los 15 y los 25 años de edad (Figura 55). El resto de intervalos de edad tienen un número de participantes similar.

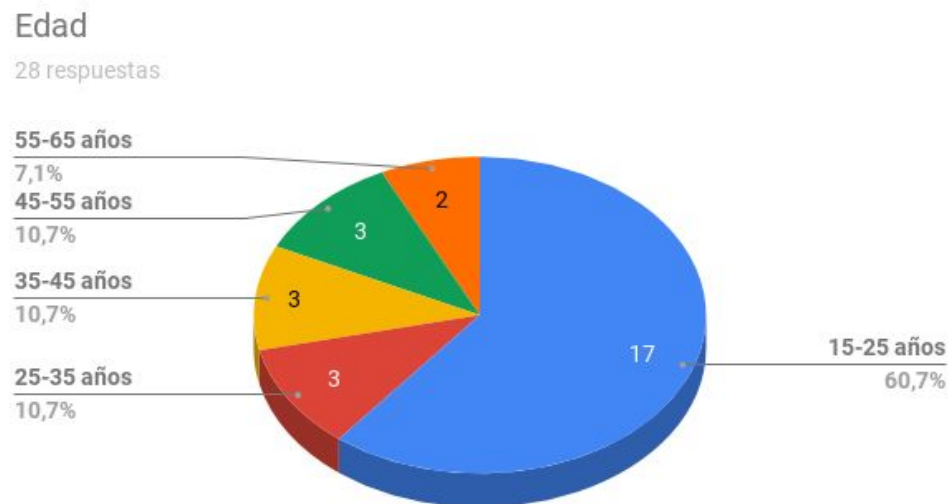


Figura 55. Edad de los encuestados

- Sexo:

Más de la mitad de participantes confirmaron ser hombres, el resto mujeres (Figura 56).

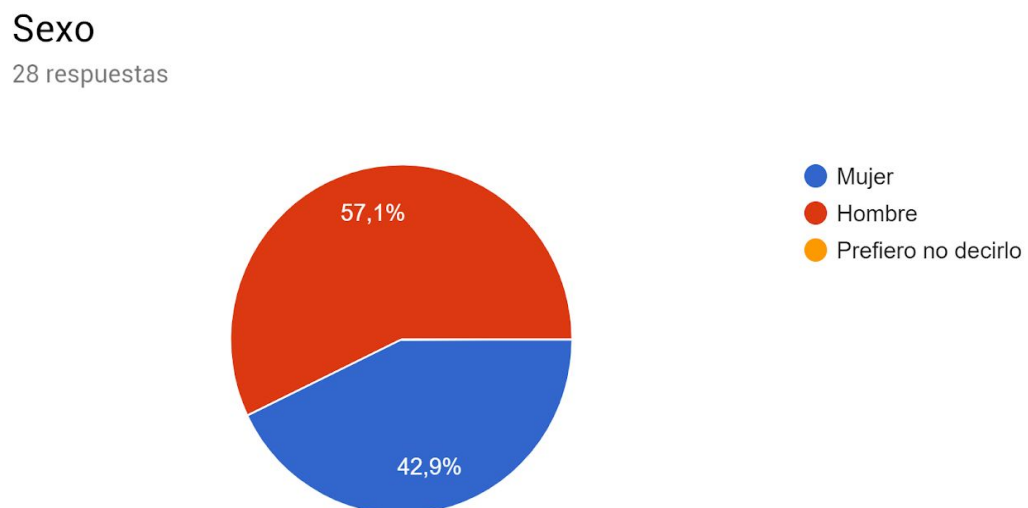


Figura 56. Sexo de los encuestados

- ¿Trabaja en la enseñanza?:

Sólo el 14,3% de los encuestados afirmó trabajar en la enseñanza o algo similar (Figura 57).

### ¿Trabajas en la enseñanza o similares?

28 respuestas

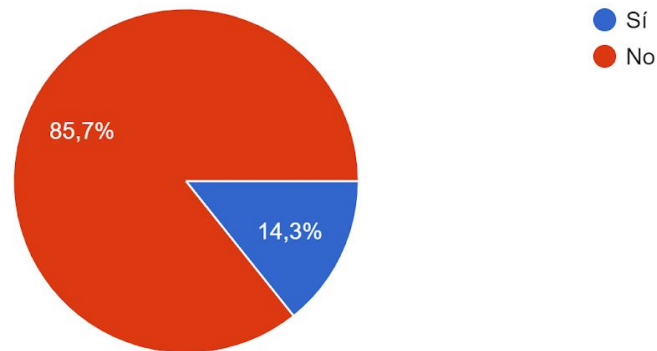


Figura 57. Respuestas a “¿Trabajas en la enseñanza o similares?”

- Nivel de satisfacción (1 muy bajo, 5 muy alto) con la primera tarea (inicio de sesión):

Más de la mitad de los participantes marcó con un 5 (muy alto) el nivel de satisfacción al iniciar sesión (Figura 58). El resto se reparten entre el 4 (28,6%) y el 3 (14,3%).

1. Accede a <http://aexamen.epizy.com>, inicia sesión con el usuario `aexamen@gmail.com` y contraseña `test`.

### Indica tu nivel de satisfacción con el inicio de sesión

28 respuestas

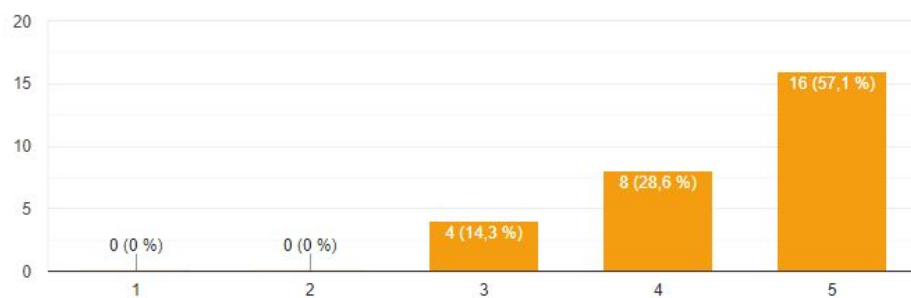


Figura 58. Satisfacción con la primera tarea

- Nivel de satisfacción (1 muy bajo, 5 muy alto) con la segunda tarea (cambio de nombre):

El 39,3% de los encuestados marcó con un 5 el nivel de satisfacción para la tarea de cambiar el nombre de su usuario (Figura 59). Seguidamente otro 35,7% contestó con un 4 a esta tarea. El resto contestó con un 3.

## 2. Cambia el nombre de tu usuario por el de algún familiar o amigo

### Indica tu nivel de satisfacción con el cambio de nombre

28 respuestas

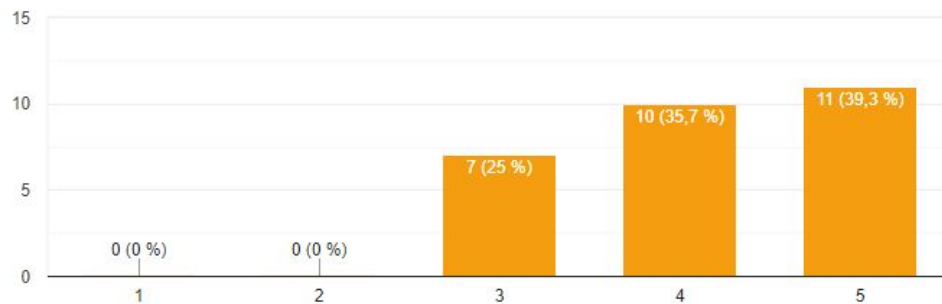


Figura 59. Satisfacción con la segunda tarea

- Nivel de satisfacción (1 muy bajo, 5 muy alto) con la tercera tarea (crear una pregunta):

La mayoría de las contestaciones se encuentran entre el 3, el 5 y el 4, siendo esta última opción la más votada (Figura 60). Existen dos votos marcando el nivel de satisfacción en 2 y uno último marcándolo en 1 (muy bajo).

### 3. Mira las asignaturas disponibles, escoge una de ellas y crea una pregunta

#### Indica tu nivel de satisfacción creando una pregunta

28 respuestas

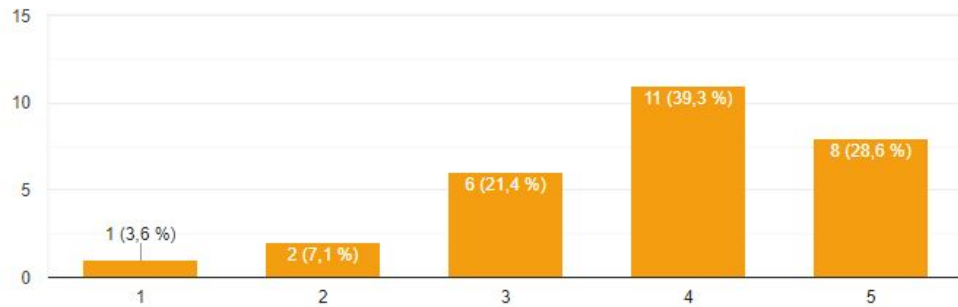


Figura 60. Satisfacción con la tercera tarea

- Nivel de satisfacción (1 muy bajo, 5 muy alto) con la cuarta tarea (crear un examen):

La satisfacción al crear un examen (Figura 61) obtuvo del 39,3% de los resultados un 4. Seguidamente el 25% contestó con un 3 y el 21,4% con un 2. Hubo un 10,7% que votó al nivel de satisfacción con un 5 (muy alto) y un último voto con nivel 1 (muy bajo).

### 4. Vuelve a las asignaturas disponibles, escoge una de ellas y crea un examen de dicha asignatura incluyendo la pregunta que creaste en el paso 3. Después guarda el examen

#### Indica tu nivel de satisfacción creando un examen

28 respuestas

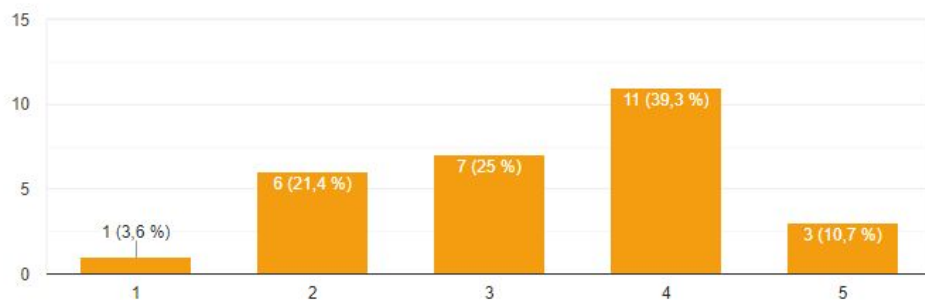


Figura 61. Satisfacción con la cuarta tarea

- Nivel de satisfacción (1 muy bajo, 5 muy alto) con la quinta tarea (generar examen PDF):

Ningún participante votó esta tarea con el mínimo de puntuación. La opción más votada es el 3 de nivel de satisfacción con un 32,1% de los votos (Figura 62) de esta pregunta. Un 25% de los votos marcaron un 4 y del resto, la mitad votó 5 y la otra mitad 2.

5. Escoge un examen de cualquier asignatura y genera el pdf que llevarás a clase el día del examen.

Indica tu nivel de satisfacción generando el pdf de un examen

28 respuestas

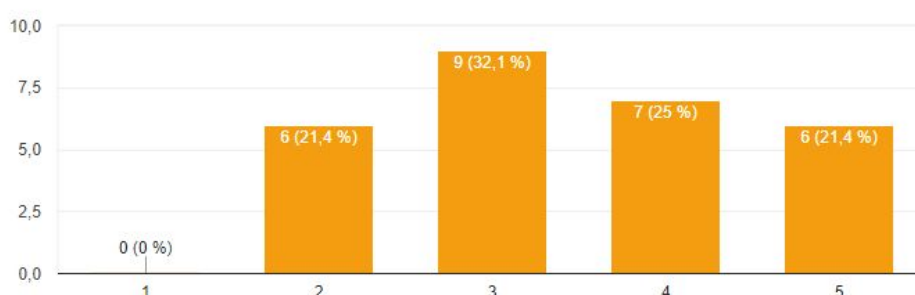


Figura 62. Satisfacción con la quinta tarea

- Tareas que han resultado más difíciles para el usuario:

La primera tarea es la que más fácil ha resultado a los participantes según la gráfica de la figura 63. Así es con la segunda y la tercera tarea. La cuarta y la quinta tarea resultan haber sido de dificultad media para la mayoría de participantes. Sin embargo, es en estas dos últimas en las que hay más personas las han encontrado difíciles que personas que las han encontrado fáciles.

¿Qué es lo que más te ha costado hacer?

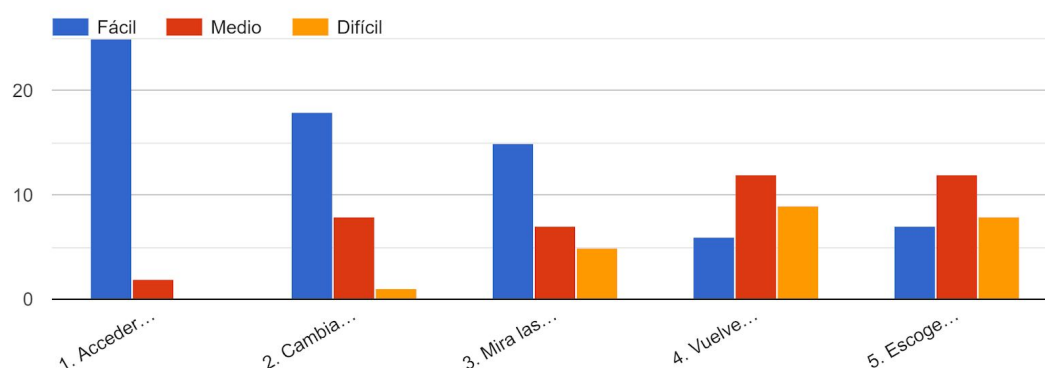


Figura 63. Diagrama con las valoraciones de las tareas más difíciles por los usuarios

- Valoraciones generales de usabilidad, colores y menús de la aplicación:

En cuanto a usabilidad la mayoría de votos marcan un 4 (Figura 64) después de haber realizado las tareas. Los colores han obtenido dos grandes grupos de personas valorando con un 3 y un 5 respectivamente. En cuanto a los menús la gran mayoría ha votado un 4, seguido por un 5 (casi la mitad de los que votaron un 4).

¿Qué te ha parecido la página web?

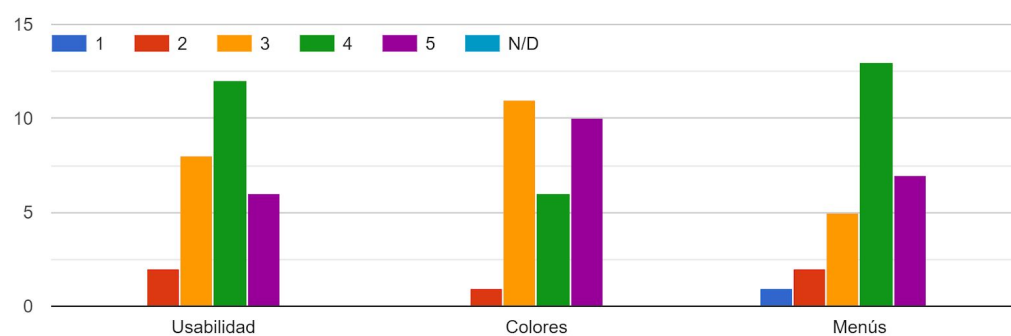


Figura 64. Diagrama con valoraciones de usabilidad, colores y menús por los usuarios

## 8.3 Muestra

El número total de respuestas es 28. Estas respuestas provienen de perfiles variados cómo se puede observar en los resultados de la primera parte del cuestionario. Además los participantes no han sido asistidos de ninguna manera mientras realizaban el cuestionario, de tal forma que los resultados fueran fiables.

## 9 Conclusiones y trabajo futuro

En este capítulo se explicarán las conclusiones del proyecto, las aportaciones realizadas por cada miembro del equipo y las líneas de trabajo futuro en la aplicación web.

### 9.1 Conclusiones

En el proyecto se han alcanzado los objetivos propuestos. Se ha conseguido desarrollar una aplicación mediante la cual se pueden gestionar los exámenes y preguntas de diversas asignaturas. También se ha conseguido una distinción clara entre distintos roles, quedando bien definidas las acciones disponibles y no disponibles para cada rol. Se ha diseñado e implementado una base de datos que mantiene la coherencia y la integridad de sus datos. Se ha logrado generar exámenes con el mismo formato oficial que se usa actualmente en la Universidad Complutense de Madrid.

Se ha logrado el objetivo de automatizar la generación de exámenes, que anteriormente se realizaba de forma manual. También se ha logrado que los profesores que imparten una asignatura tengan más información y puedan ver las preguntas y exámenes creados por sus compañeros.

Una posible desventaja es que se requiere un manejo básico de ofimática por parte de todos los profesores que imparten la asignatura. Otra desventaja es que se requiere conexión a internet para poder usar la plataforma.

### 9.2 Aportaciones

A continuación se presentan las aportaciones realizadas por cada uno de los participantes en el proyecto.

#### 9.2.1 Daniel Fernández Carnero

Las aportaciones de Daniel han sido las siguientes:

- Investigación de templates CSS en la web y adaptación de uno de ellos a la aplicación.
- Diseño e implementación del formulario de login y validación de los campos.
- Diseño de la página “preguntas”, implementación de las funcionalidades de creación, edición y eliminación de preguntas junto con la validación de todos los formularios asociados. Diseño e implementación de la vista detallada de una pregunta en concreto, incluyendo la tabla con todas las modificaciones de dicha pregunta.
- Diseño y funcionalidad de la barra de navegación. Implementación del cierre de sesión y acceso al perfil propio.
- Diseño de la página “perfil propio”, implementación de las funcionalidades de cambio de los diversos campos y validación de los formularios.

### 9.2.2 Samuel Javier García Moreno

Las aportaciones de Samuel han sido las siguientes:

- Diseño de las páginas “registro” y “olvidé mi contraseña” e implementación de ambos formularios junto con su validación.
- Diseño e implementación de los filtros y barra de búsqueda presentes en muchas de las páginas de la aplicación.
- Diseño de la página “exámenes” e implementación de las funcionalidades de gestión de exámenes. Diseño e implementación de la página de creación de exámenes. Diseño e implementación del formulario para generar exámenes y la funcionalidad con FPDF.
- Diseño e implementación del panel de control, las funcionalidades de descarga, visualización y eliminación del log y la tabla de peticiones de registro.

### 9.3 Líneas de trabajo futuro

Pese a que se han alcanzado los objetivos propuestos hay varias líneas de trabajo futuro que se podrían tomar. A continuación se expondrán algunas de ellas:

- Mostrar en los detalles de una pregunta, un listado detallado de en qué exámenes se está usando la pregunta.
- Implementar la adaptación de la aplicación a pantallas más pequeñas y a pantallas móviles.
- Desarrollar una aplicación móvil que conecte con la misma base de datos que la aplicación web, de esta manera compartir los datos.
- Implementar, en la vista de definir parámetros de examen y en generar examen, la opción para el usuario de subir una imagen con un escudo propio para que se muestre en la cabecera del examen.
- Traducir toda la aplicación al inglés, ya que es la lengua franca actualmente en el mundo académico, y que la aplicación se pueda alternar entre los dos de forma cómoda.



## 9 Conclusions and future work

The following chapter will explain the project conclusions, the contributions made by each member of the team and the future lines of work in the web application.

### 9.1 Objectives

The goals proposed in the beginning have been achieved. An application which manages exams and questions from multiple subjects has been developed correctly. In addition, differentiating the roles permissions and aspects has been successfully done. A fully coherent and data integrity proven database has been designed and developed. Finally, generating exams with the exact same aspect, as UCM teachers would manually make, has also been achieved.

The target of automating the generation of exams, which was previously done manually, has been achieved. It has also made it possible for teachers who teach a subject to have more information and to be able to see the questions and exams created by their colleagues.

A possible disadvantage is that it requires basic computer management by all teachers who teach the subject. Another disadvantage is that an Internet connection is required in order to use the platform the way we propose.

### 9.2 Contribution

Below are the contributions made by each of the participants in the project. Both project members have participated equally on the application's design and features implementation. However, each of them has dived into some aspects deeper than the others.

#### 9.2.1 Daniel Fernández Carnero

Daniel's contributions were as follows:

- Research of CSS templates in the web and adaptation of one of them to the application.
- Design and implementation of the login form and validation of the fields.
- Design of the page "questions". Questions creation, edition and elimination functionalities implementation together with the validation of all the associated forms. Design and implementation of the detailed view for a specific question, including the table with all of the historic modifications of the question.
- Design and functionality of the navigation bar. Implementation of logout and access to own profile functionalities.
- Design of the "self-profile" page, implementation of the functionalities for changing different fields and validation of the forms.

### 9.2.1 Samuel Javier García Moreno

Samuel's contributions were as follows:

- Design of the pages "registration" and "forgot my password" and implementation of both forms together with their validation.
- Design and implementation of the filters and search bar present in many of the pages of the application.
- Design of the "exams" page and implementation of the exams management functionalities. Design and implementation of the exam creation page. Design and implementation of the form for generating exams and its functionality using FPDF library.
- Design and implementation of the control panel, the functionalities for downloading, visualizing and eliminating the log and the table of registration requests.

### 9.3 Future work lines

Although the proposed objectives have been achieved, there are several lines of future work that could be taken. The following are some of them:

- Show in the details of a question, a detailed listing of which exams is the question present in.
- Implement the adaptation of the application to smaller and mobile screens.
- Develop a mobile application that connects to the same database as the web application, while sharing the data.
- Implement, in the defining exam parameters and generating exams views, the functionality so that it gives the user the option to upload an image with its own shield to be displayed in the exam header.
- Translate the entire application into English, since it is the most widely used language in the academic world, and the application could alternate between both in a comfortable way.

# Bibliografía

[1] Google Classroom. (2019). *Administra la enseñanza y el aprendizaje con classroom | Google for Education*. [online] Disponible en: [https://edu.google.com/intl/es-419/products/classroom/?modal\\_active=none](https://edu.google.com/intl/es-419/products/classroom/?modal_active=none) [Accedido 19 May 2019].

[2] Moodle (2019). *Características de Moodle 3.6 - MoodleDocs*. [online] Disponible en: [https://docs.moodle.org/all/es/Caracter%C3%ADsticas\\_de\\_Moodle\\_3.6](https://docs.moodle.org/all/es/Caracter%C3%ADsticas_de_Moodle_3.6) [Accedido 23 May 2019].

[3] Blackboard (2019). *Blackboard | Servicios y tecnología para la educación*. [online] Disponible en: <https://es.blackboard.com/index.html?nog=1&cc=US> [Accedido 23 May 2019].

[4] Edmodo. (2019). *Edmodo*. [online] Disponible en: <https://www.edmodo.com/?language=es> [Accedido 23 May 2019].

[5] Exelearning.net. (2019). *Características de eXeLearning | eXeLearning.net*. [online] Disponible en: <http://exelearning.net/caracteristicas/> [Accedido 23 May 2019].

[6] Sublimetext.com. (2019). *Sublime Text - A sophisticated text editor for code, markup and prose*. [online] Disponible en: <https://www.sublimetext.com/> [Accedido 23 May 2019].

[7] Php.net. (2019). *PHP: ¿Qué es PHP? - Manual*. [online] Disponible en: <https://php.net/manual/es/intro-what-is.php> [Accedido 23 May 2019].

[8] Fpdf.org. (2019). *FPDF*. [online] Disponible en: <http://www.fpdf.org/> [Accedido 19 May 2019].

[9] GitHub. (2019). *PHPMailer/PHPMailer*. [online] Disponible en: <https://github.com/PHPMailer/PHPMailer/blob/master/README.md> [Accedido 19 May 2019].

[10] Documentación web de MDN. (2019). *JavaScript*. [online] Disponible en: <https://developer.mozilla.org/es/docs/Web/JavaScript> [Accedido 23 May 2019].

[11] Bootstrap, Mark Otto, a. (2019). *Bootstrap*. [online] Getbootstrap.com. Disponible en: <https://getbootstrap.com/> [Accedido 19 May 2019].

[12] js.foundation, J. (2019). *jQuery*. [online] JQuery.com. Disponible en: <https://jquery.com/> [Accedido 19 May 2019].

[13] js.foundation, J. (2019). *jQuery.ajax()* | *jQuery API Documentation*. [online] Api.jquery.com. Disponible en: <http://api.jquery.com/jquery.ajax/> [Accedido 19 May 2019].

[14] Tempusdominus.github.io. (2019). *Tempus Dominus - Bootstrap 4*. [online] Disponible en: <https://tempusdominus.github.io/bootstrap-4/> [Accedido 20 May 2019].

[15] CSS - MDN Web Docs. (2019). *CSS: Cascading Style Sheets*. [online] Disponible en: <https://developer.mozilla.org/en-US/docs/Web/CSS> [Accedido 23 May 2019].

[16] HTML - Documentación web de MDN. (2019). *HTML*. [online] Disponible en: <https://developer.mozilla.org/es/docs/Web/HTML> [Accedido 23 May 2019].

[17] Mysql.com. (2019). *MySQL*. [online] Disponible en: <https://www.mysql.com/> [Accedido 19 May 2019].

[18] Json.org. (2019). *JSON*. [online] Disponible en: <https://www.json.org/> [Accedido 19 May 2019].

[19] Documentación web de MDN. (2019). *¿Que es un servidor WEB?* [online] Disponible en: [https://developer.mozilla.org/es/docs/Learn/Common\\_questions/Que\\_es\\_un\\_servidor\\_WEB](https://developer.mozilla.org/es/docs/Learn/Common_questions/Que_es_un_servidor_WEB) [Accedido 23 May 2019].

[20] Group, D. (2019). *Documentation: Apache HTTP Server - The Apache HTTP Server Project*. [online] Httpd.apache.org. Disponible en: <http://httpd.apache.org/docs/> [Accedido 23 May 2019].

[21] Php.net. (2019). *PHP: filemtime - Manual*. [online] Disponible en: <https://www.php.net/manual/es/function.filemtime.php> [Accedido 19 May 2019].

[22] Book.cakephp.org. (2019). *¿Qué es CakePHP y por qué hay que utilizarlo?* [online] Disponible en: <https://book.cakephp.org/1.3/es/The-Manual/Beginning-With-CakePHP/What-is-CakePHP-Why-Use-it.html> [Accedido 23 Jun. 2019].

[23] Atom. (2019). *A hackable text editor for the 21st Century*. [online] Disponible en: <https://atom.io/> [Accedido 23 Jun. 2019].

# ANEXOS

## Anexo I: Guía de uso

La aplicación posee dos roles de usuario, por lo tanto dependiendo del rol se podrá realizar diferentes acciones. En ambos casos la primera acción es realizar el login. Se debe introducir un correo electrónico y una contraseña (Figura 65) y dependiendo del rol de la cuenta se accede a una interfaz diferente.

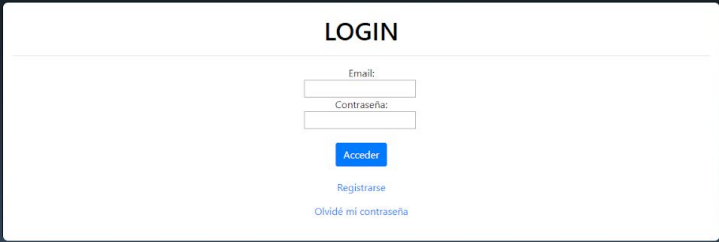
The image shows a login form titled "LOGIN" centered on a dark blue background. The form is a white rectangle containing the following elements: a label "Email:" above a text input field, a label "Contraseña:" above another text input field, a blue button labeled "Acceder", and two links below the button: "Registrarse" and "Olvidé mi contraseña".

Figura 65. Login

Es posible restablecer la contraseña de la cuenta (Figura 66). Para ello se pide introducir el correo electrónico del usuario que quiere restablecer la contraseña de su cuenta (la cual no recuerda). Una vez enviado, recibirá un correo electrónico con un enlace para proceder a restablecer su contraseña correctamente.

The image shows a form titled "RESTABLECER CONTRASEÑA" centered on a dark blue background. The form is a white rectangle containing the following elements: a subtitle "Introduce tu correo electrónico para poder restablecer la contraseña mediante el correo que te enviaremos:", a label "Email:" above a text input field containing "email@example.com", a blue button labeled "Enviar", and two links below the button: "Iniciar sesión" and "Registrarse".

Figura 66. Restablecer contraseña

A continuación se van a mostrar las funcionalidades del sistema de acuerdo al rol del usuario.

## 1 Rol profesor

### 1.2 Página principal

Si se ha accedido como profesor aparecerá una pantalla con dos accesos (Figura 67), uno para ver las asignaturas a las que está asignado el profesor y otro para ver todos los exámenes de las asignaturas a las que está asignado.

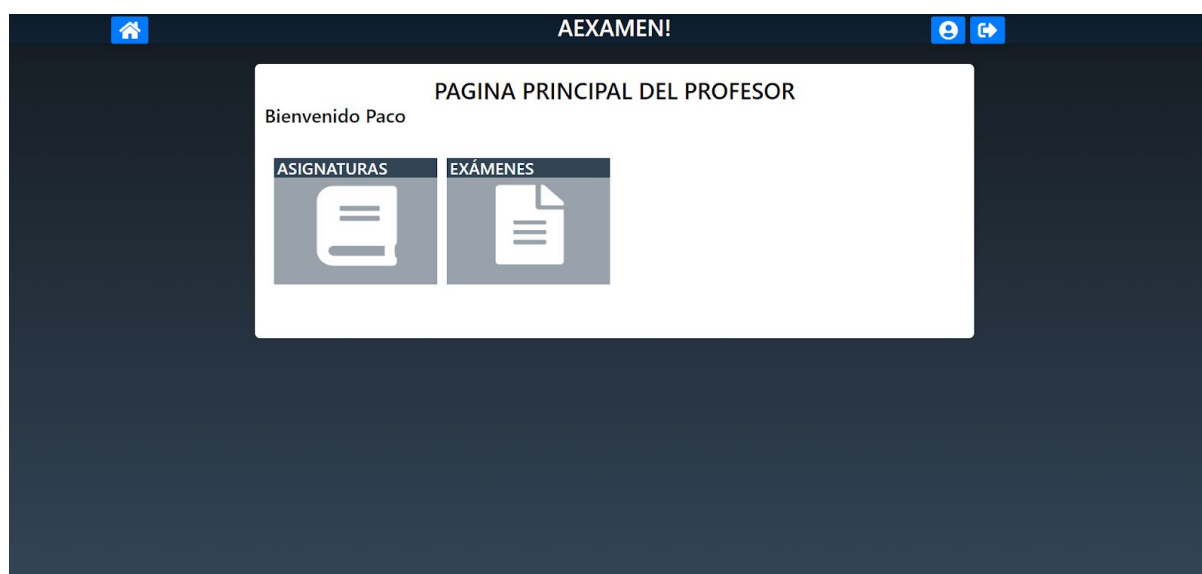


Figura 67. Página principal

### 1.3 Exámenes de todas las asignaturas

Si se pulsa sobre exámenes, se mostrarán todos los exámenes de las asignaturas a las cuales esté asignado el profesor (Figura 68). En la tabla en la que se muestran los exámenes se podrá ver información acerca de estos. Así mismo, podrá filtrar por asignatura, por autor y hacer uso de una barra de búsqueda y crear un examen de la asignatura que se seleccione en el desplegable y pulsando el botón con el símbolo “+”. Además, por cada examen, podrá ver más detalles de este, generarlo, editarlo o borrarlo.



Figura 68. Exámenes del profesor

## 1.4 Crear examen

En esta pantalla (Figura 69) se tiene que definir el nombre y las preguntas que tendrá el examen antes de guardarlo. El nombre se define en el campo rellenable situado en la parte superior. Para seleccionar las preguntas, se pulsa el botón con el símbolo “+” adyacente al tema, una vez pulsado se despliega un *pop-up* con las preguntas asignadas a ese tema y una vez seleccionada se pulsa en el botón “Añadir”. Cuando ya está añadida la pregunta se puede definir cuántos puntos vale la pregunta usando las flechas que están a ambos lados de los puntos de la pregunta. Por defecto cuando se añade una pregunta vale 1 punto.

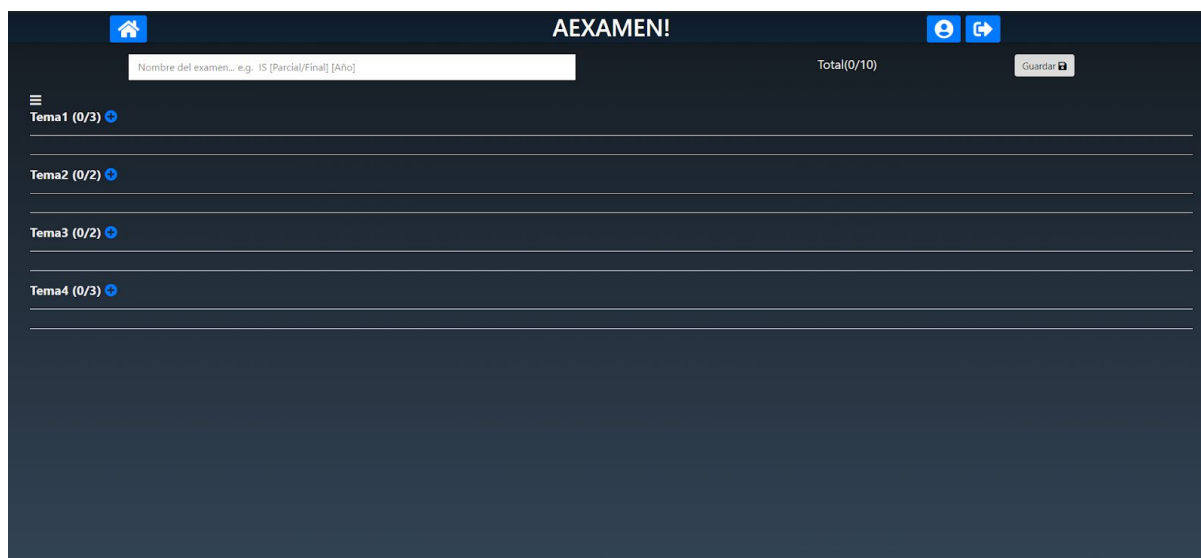
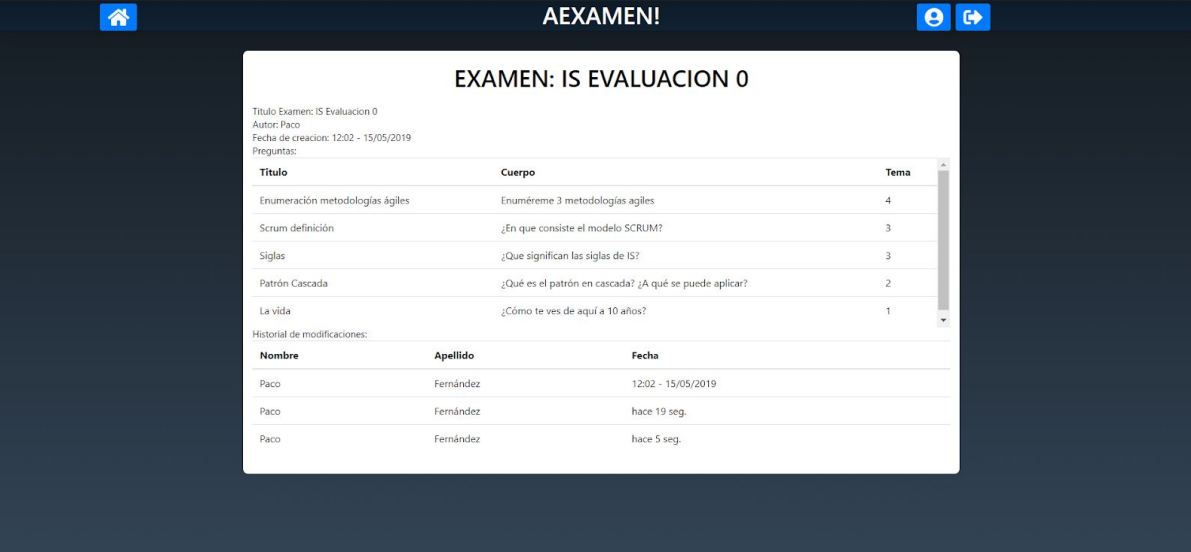


Figura 69. Crear examen

## 1.5 Detalles de examen

En esta pantalla (Figura 70) se puede observar, además del título, autor y fecha de modificación, qué preguntas tiene el examen y el historial de modificaciones que ha sufrido, teniendo en esta última tanto el autor como la fecha de la modificación.



**EXAMEN: IS EVALUACION 0**

Título Examen: IS Evaluacion 0  
Autor: Paco  
Fecha de creación: 12:02 - 15/05/2019  
Preguntas:

Título	Cuerpo	Tema
Enumeración metodologías ágiles	Enuméreme 3 metodologías ágiles	4
Scrum definición	¿En que consiste el modelo SCRUM?	3
Siglas	¿Que significan las siglas de IS?	3
Patrón Cascada	¿Qué es el patrón en cascada? ¿A qué se puede aplicar?	2
La vida	¿Cómo te ves de aquí a 10 años?	1

Historial de modificaciones:

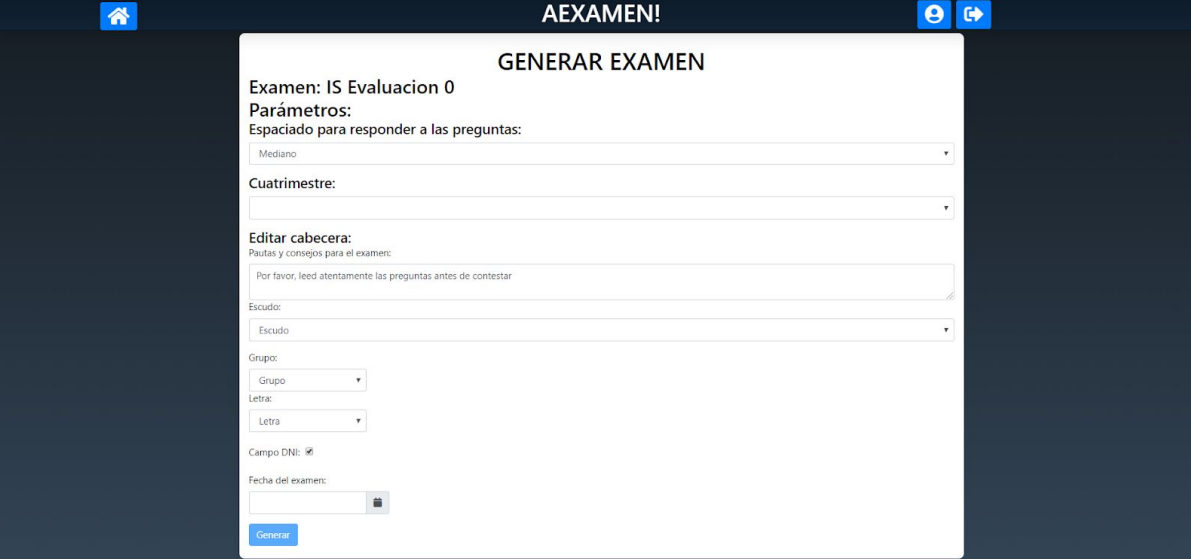
Nombre	Apellido	Fecha
Paco	Fernández	12:02 - 15/05/2019
Paco	Fernández	hace 19 seg.
Paco	Fernández	hace 5 seg.

Figura 70. Detalles de examen

## 1.6 Generar examen

En esta pantalla (Figura 71) se tienen que seleccionar diversos parámetros que va a tener el examen. Algunos de ellos vienen definidos por defecto por el coordinador de la asignatura.

Una vez que se hayan definido los parámetros de la asignatura se puede generar el examen.



**GENERAR EXAMEN**

Examen: IS Evaluacion 0

**Parámetros:**

Espaciado para responder a las preguntas:  
Mediano

Cuatrimestre:

**Editar cabecera:**  
Pautas y consejos para el examen:  
Por favor, leed atentamente las preguntas antes de contestar

Escudo:  
Escudo

Grupo:  
Grupo

Letra:  
Letra

Campo DNI: ☒

Fecha del examen:

**Generar**

Figura 71. Generar examen



## 1.7 Editar examen

En esta pantalla (Figura 72) se podrá cambiar el nombre al examen, modificar los puntos que tiene una pregunta y añadir y quitar preguntas. Una vez que se hayan realizado los cambios que se ha querido añadir se debe pulsar el botón guardar para salvarlos.

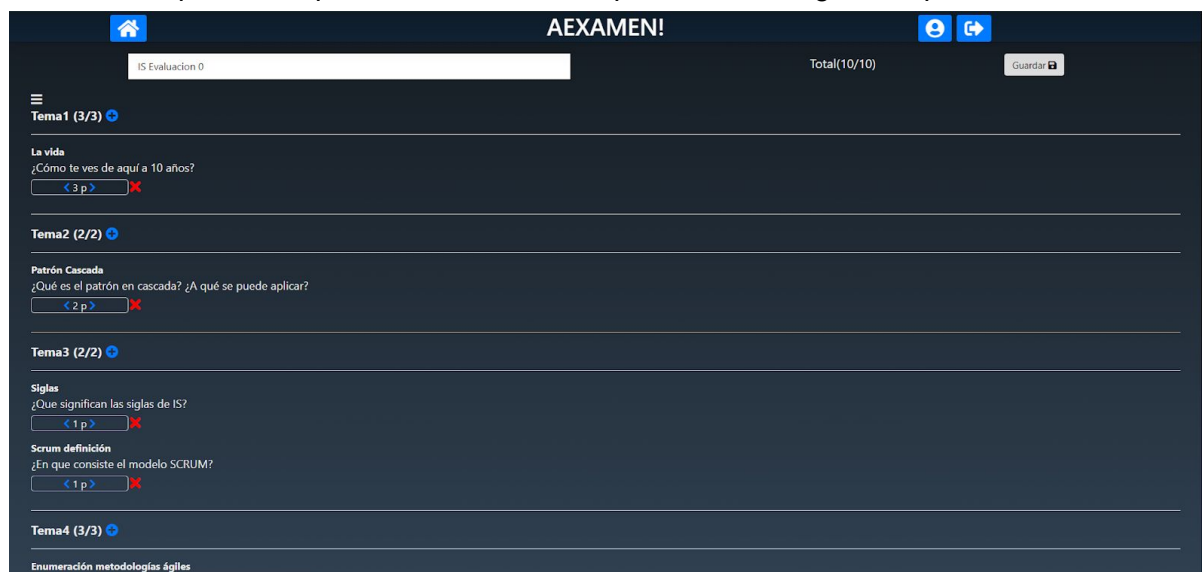


Figura 72. Editar examen

## 1.8 Borrar examen

En esta pantalla (Figura 73) se borra el examen siempre y cuando confirmemos nuestra decisión en los dos cuadros de diálogo que se muestran, de esta forma se evitan errores a la hora de borrar exámenes.

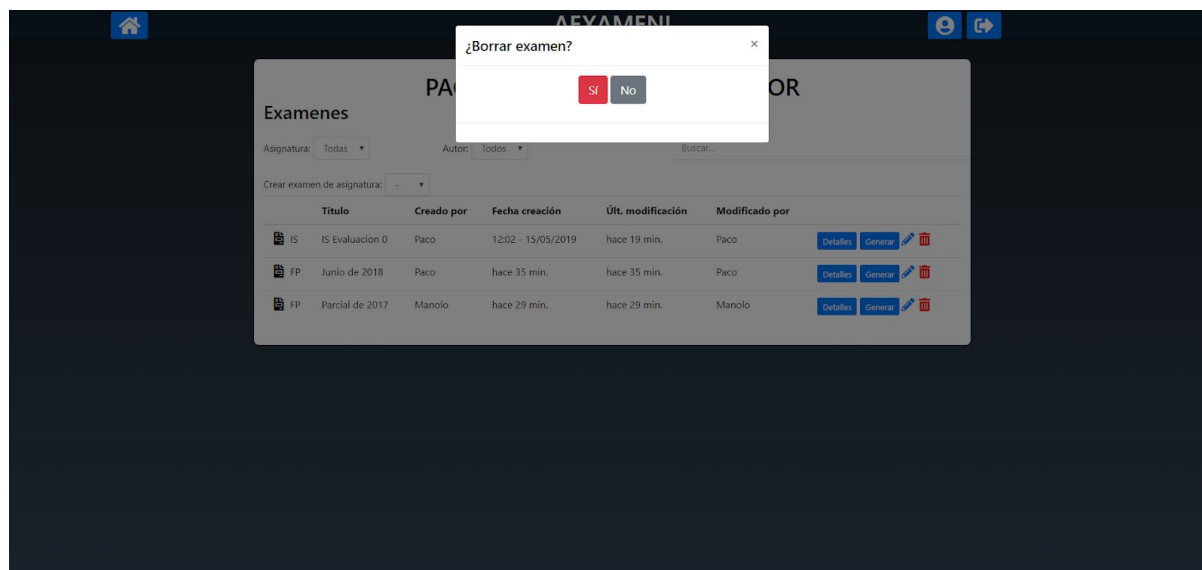


Figura 73. Borrar examen

## 1.9 Asignaturas

En esta vista (Figura 74) se muestran al usuario las asignaturas de las cuales forma parte (ya sea profesor que imparte o coordinador). Puede filtrar entre estas usando la barra de búsqueda u ordenarlas haciendo click sobre cualquier columna.



Siglas	Nombre asignatura	Coordinador
IS	Ingeniería del software	Si
FP	Fundamentos de la programación	No
DR	Derecho Romano	Si

Figura 74. Asignaturas de un profesor

## 1.10 Asignatura

Cada vista (Figura 75) de una asignatura permite acceder a las preguntas o exámenes de dicha asignatura.



Figura 75. Asignatura

## 1.11 Preguntas

En esta vista (Figura 76) se muestran todas las preguntas de la asignatura seleccionada. Se pueden filtrar por autor, hacer búsqueda de algo en específico u ordenar

por cualquier columna de las que se muestran. Para crear nuevas preguntas se pulsa en el “+”. Para cada pregunta se puede ver los detalles, editarla o eliminarla.

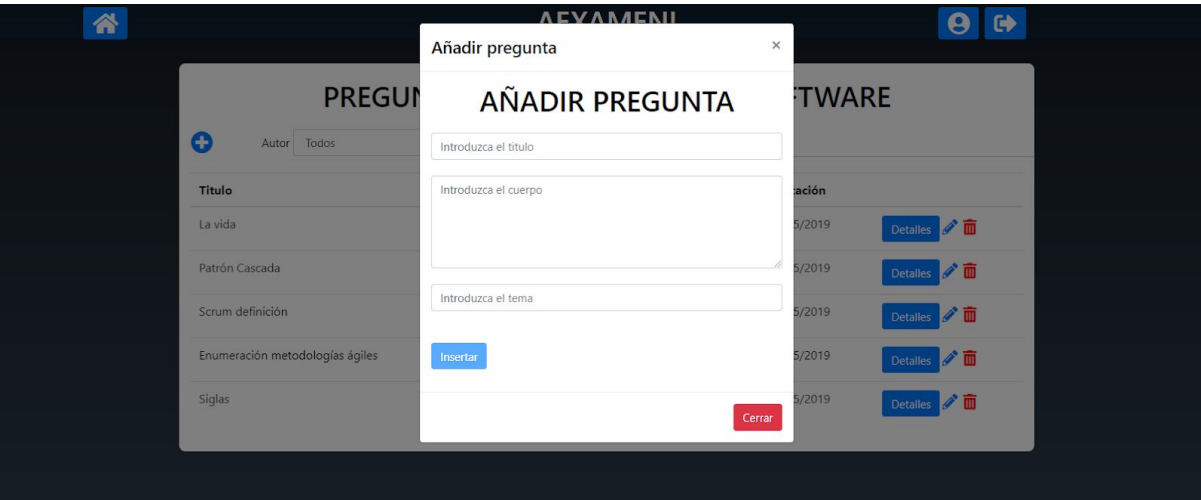


Título	Tema	Autor	Fecha creación	Últ. modificación	
La vida	1	Paco	19:53 - 07/01/2019	21:38 - 05/05/2019	Detalles  
Patrón Cascada	2	Paco	17:20 - 14/05/2019	17:20 - 14/05/2019	Detalles  
Scrum definición	3	Paco	21:17 - 17/05/2019	21:17 - 17/05/2019	Detalles  
Enumeración metodologías ágiles	4	Paco	21:18 - 17/05/2019	21:18 - 17/05/2019	Detalles  
Siglas	3	Alex	21:00 - 17/05/2019	21:16 - 17/05/2019	Detalles  

Figura 76. Preguntas de asignatura

## 1.12 Crear pregunta

Para crear una pregunta hay que completar el formulario rellenando título, cuerpo y tema (Figura 77). Hasta que no se completen los tres campos no se podrá hacer click en “Insertar”, lo que hará que se añada la nueva pregunta.



**AÑADIR PREGUNTA**

Introduzca el título

Introduzca el cuerpo

Introduzca el tema

Insertar

Cerrar

Figura 77. Añadir pregunta

## 1.13 Detalles de pregunta

En la vista (Figura 78) de detalle de una pregunta se observa el título, cuerpo y tema de la pregunta. También aparece el autor, la fecha de creación y un historial de modificaciones que identifica el nombre, apellido y fecha de modificación por cada una.



Figura 78. Detalles de pregunta

### 1.14 Editar pregunta

Para editar una pregunta, sólo hace falta completar el campo el cual se vaya a editar (Figura 79), pudiendo dejar los demás sin tocar. Una vez acabado, se pulsa sobre "Editar" y se hace efectivo el cambio.

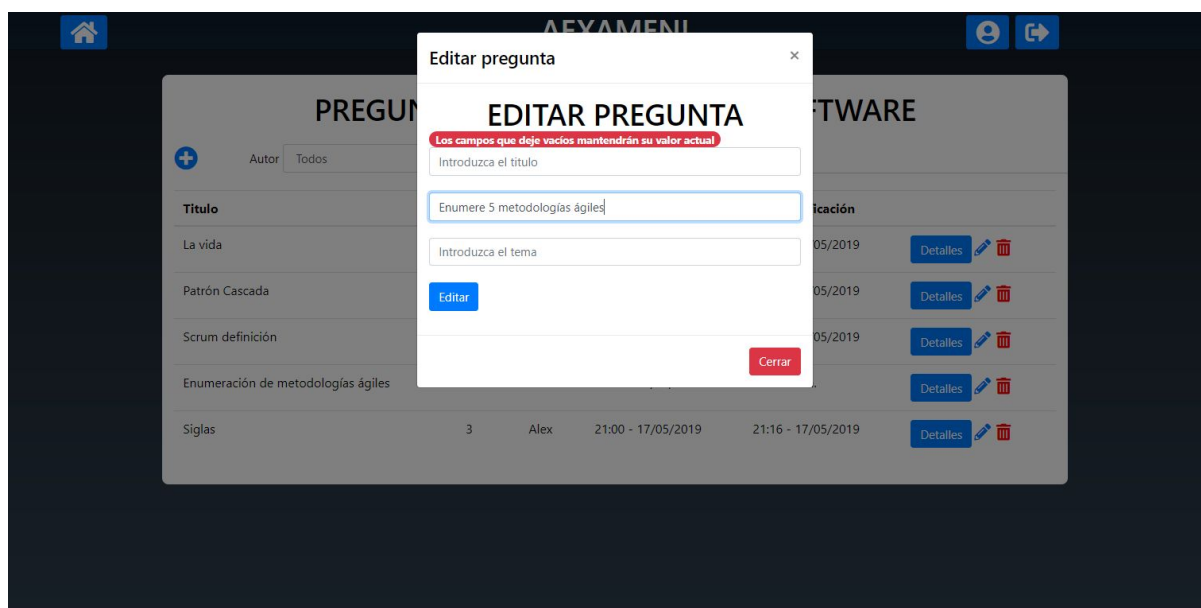


Figura 79. Editar pregunta

### 1.15 Borrar pregunta

En esta pantalla (Figura 80) se borra la pregunta siempre y cuando se confirme la decisión en los dos cuadros de diálogo que se muestran, de esta forma se evitan errores inesperados a la hora de borrar preguntas. En caso de que la pregunta la haya creado otro usuario o aparezca en uno o más exámenes no podrá ser borrada.

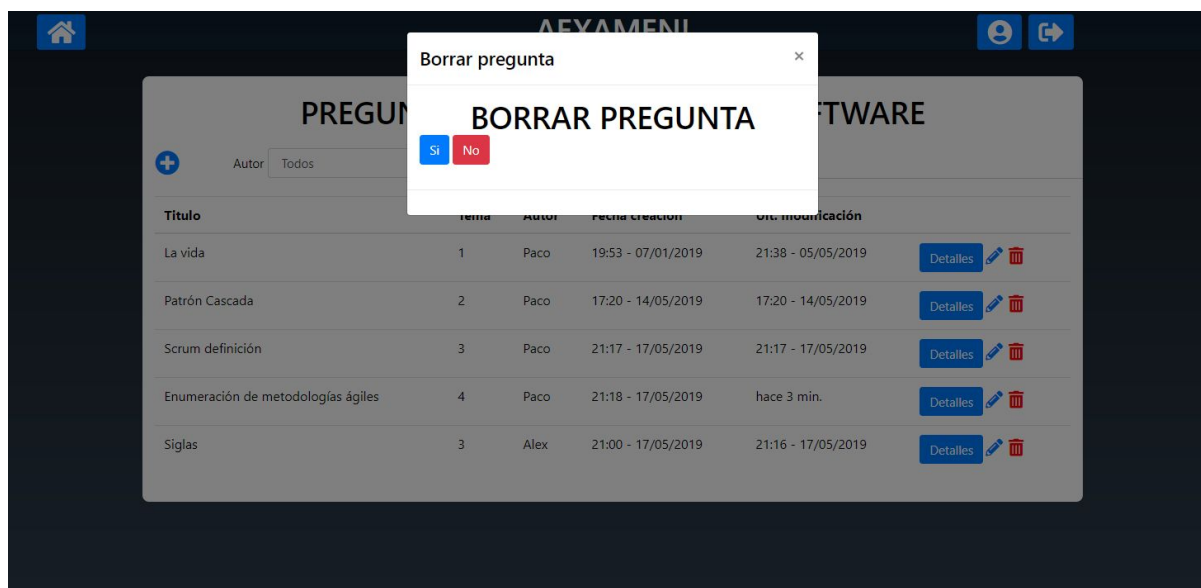


Figura 80. Borrar pregunta

### 1.16 Perfil propio

Esta vista (Figura 81), la cual es accesible desde la barra superior en todo momento una vez se ha iniciado sesión, muestra todos los detalles del perfil de un usuario. En ella se puede modificar el nombre, apellido y la contraseña pulsando sobre los respectivos botones.



Figura 81. Perfil propio

## 2 Rol coordinador

Este rol lo desempeña un profesor dentro de una asignatura que coordine. Por lo tanto tiene en común todas las pantallas y acciones previamente explicadas de profesor. Pero dentro de la asignatura que coordine tendrá acceso a otras pantallas y acciones. En las demás asignaturas que no sea coordinador tendrá acceso a las mismas pantallas y acciones que un profesor, pues es el rol que desempeñará en esas asignaturas.

## 2.1 Asignatura

En esta pantalla (Figura 82), además de mostrar los exámenes y las preguntas como al resto de profesores que imparten esta asignatura, tiene las opciones de profesores y parámetros. En el enlace “profesores” podrá definir qué profesores imparten la asignatura. En parámetros, podrá definir qué parámetros tienen los exámenes de esta asignatura por defecto.



Figura 82. Asignatura

## 2.2 Preguntas

Esta pantalla tiene el mismo aspecto que la pantalla de preguntas de profesores. Pero al ser coordinador de la asignatura, se podrá borrar preguntas que haya creado cualquier profesor.

## 2.3 Gestionar profesores

En esta pantalla (Figura 83) el coordinador podrá ver qué profesores tiene en la asignatura que coordina, así como detalles de estos profesores. También podrá buscar profesores con la barra de búsqueda. Para añadir profesores se tiene que pulsar el botón con el símbolo “+” y se desplegará un *pop-up*, donde aparecen todos los profesores del sistema. Una vez que se selecciona el profesor se pulsa en añadir y dicho profesor se añade a la asignatura. Para borrarlos pulsar el icono de papelera y confirmar que se quiere borrar a dicho profesor de la asignatura.



Figura 83. Profesores de una asignatura

## 2.4 Parámetros de exámenes

En la siguiente vista (Figura 84) el coordinador puede elegir, en primer lugar, el número total de temas de la asignatura, el máximo de puntos para los exámenes y la distribución de estos por cada tema. Si se aumenta o disminuye el campo “Número total de temas” se añade o elimina dinámicamente un campo de tema al listado. Además se pueden elegir el espaciado entre las preguntas y el texto inicial que verán los alumnos antes que cualquier pregunta. Cabe destacar que a la hora de generar un examen en PDF, el profesor en cuestión podrá modificar estos dos últimos parámetros a su gusto. Si en algún momento no está a gusto con la distribución de temas o cualquier otra cosa puede hacer click en el botón superior “Restablecer valores” volviendo al estado inicial sin guardar nada.

**PAGINA DE COORDINADOR** Restablecer valores

**Puntos por tema**

Numero total de temas:  Puntos por examen:  Tema 1:

Tema 2:  Tema 3:  Tema 4:

**Espaciado entre preguntas por defecto**

☐ Pequeño ☒ Medio ☐ Grande

**Texto breve inicial(consejos, normas, etc.)**

Por favor, leed atentamente las preguntas antes de contestar

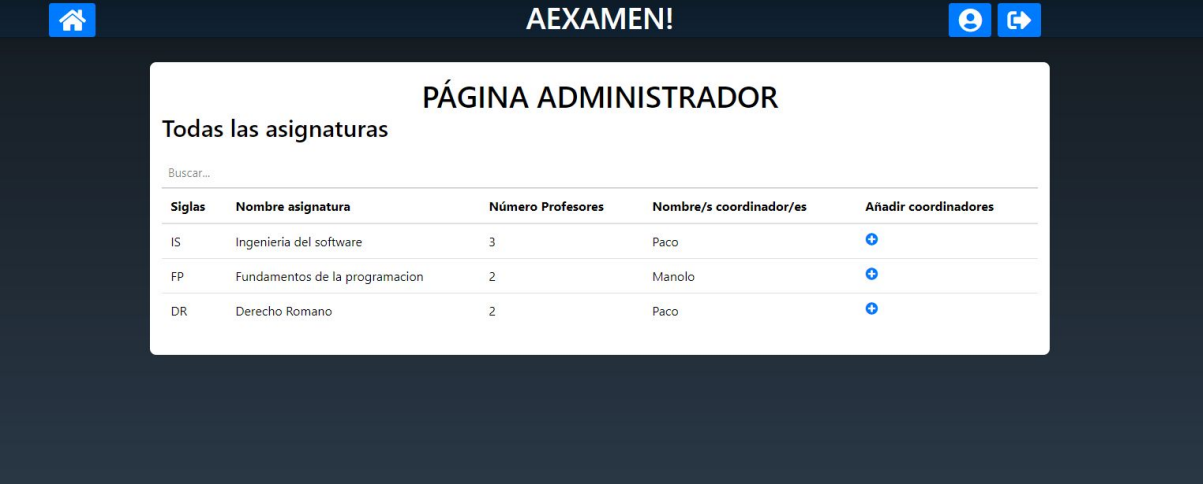
Guardar

Figura 84. Parámetros de exámenes

## 3 Rol administrador

### 3.1 Asignaturas

El administrador gestiona todas las asignaturas dadas de alta en la plataforma (Figura 85). Puede observar el número de profesores en cada asignatura, el/los nombre/s de coordinador/es. Además puede añadir o eliminar coordinadores a una asignatura haciendo click en el “+”.



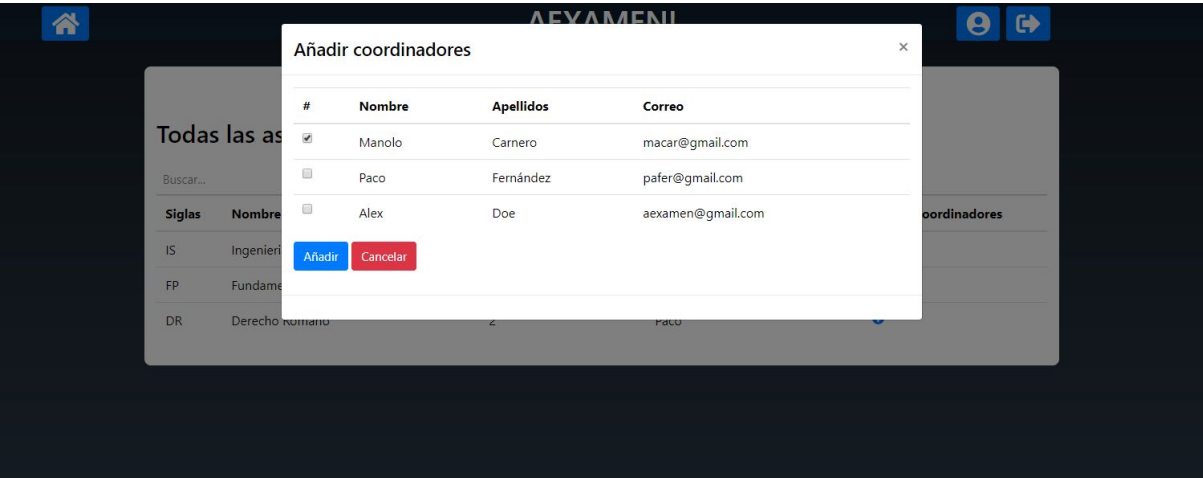
The screenshot shows the 'PÁGINA ADMINISTRADOR' interface. At the top, there's a navigation bar with a home icon, the text 'AEXAMEN!', and user profile and share icons. Below this, the title 'PÁGINA ADMINISTRADOR' is centered. Underneath, it says 'Todas las asignaturas' followed by a search bar labeled 'Buscar...'. The main content is a table with the following data:

Siglas	Nombre asignatura	Número Profesores	Nombre/s coordinador/es	Añadir coordinadores
IS	Ingeniería del software	3	Paco	
FP	Fundamentos de la programación	2	Manolo	
DR	Derecho Romano	2	Paco	

Figura 85. Asignaturas

### 3.2 Añadir coordinadores

Al añadir o eliminar coordinadores de una asignatura se debe seleccionar mediante *checkboxes* (Figura 86) el/los profesor/es que se quiera establecer para coordinar la asignatura y pulsar en “Añadir”. Nunca se dejará continuar si no se ha seleccionado ninguno.



The screenshot shows a modal window titled 'Añadir coordinadores' with a close button (X) in the top right corner. The modal contains a table with the following data:

#	Nombre	Apellidos	Correo
<input checked="" type="checkbox"/>	Manolo	Carnero	macar@gmail.com
<input type="checkbox"/>	Paco	Fernández	pafer@gmail.com
<input type="checkbox"/>	Alex	Doe	aexamen@gmail.com

At the bottom of the modal, there are two buttons: 'Añadir' (blue) and 'Cancelar' (red).

Figura 86. Añadir coordinadores a una asignatura



### 3.3 Exámenes

En la vista de exámenes (Figura 87) se muestran al administrador todos los exámenes del sistema. Para cada uno de ellos puede ver los detalles y eliminarlo (como cualquier profesor). Además se podrá filtrar por asignatura, autor y hacer búsquedas concretas.



Figura 87. Exámenes

### 3.4 Profesores del sistema

En esta pantalla (Figura 88) el administrador podrá ver la información de todos los profesores que hay dados de alta en el sistema. También podrá editar, borrar y asignar a los profesores las asignaturas que coordinarán.

Para editar se pulsa sobre el icono del lápiz, se despliega un *pop-up* con el nombre, apellidos y correo del profesor. Todos estos campos se pueden editar y una vez que estén editados se pulsa sobre el botón “Actualizar” para guardar los cambios y cancelar en caso de que no se quieran tener en cuenta.

Para borrar un usuario, basta con pulsar el icono de la papelera y aceptar los *pop-up* de confirmación que se desplegarán. Para definir qué asignaturas coordinará un profesor se pulsará sobre el botón “Asignaturas coordinadas”. Este botón desplegará un *pop-up* en el que se muestran todas las asignaturas que hay en el sistema, apareciendo primero y seleccionadas las que el profesor está coordinando en ese momento.

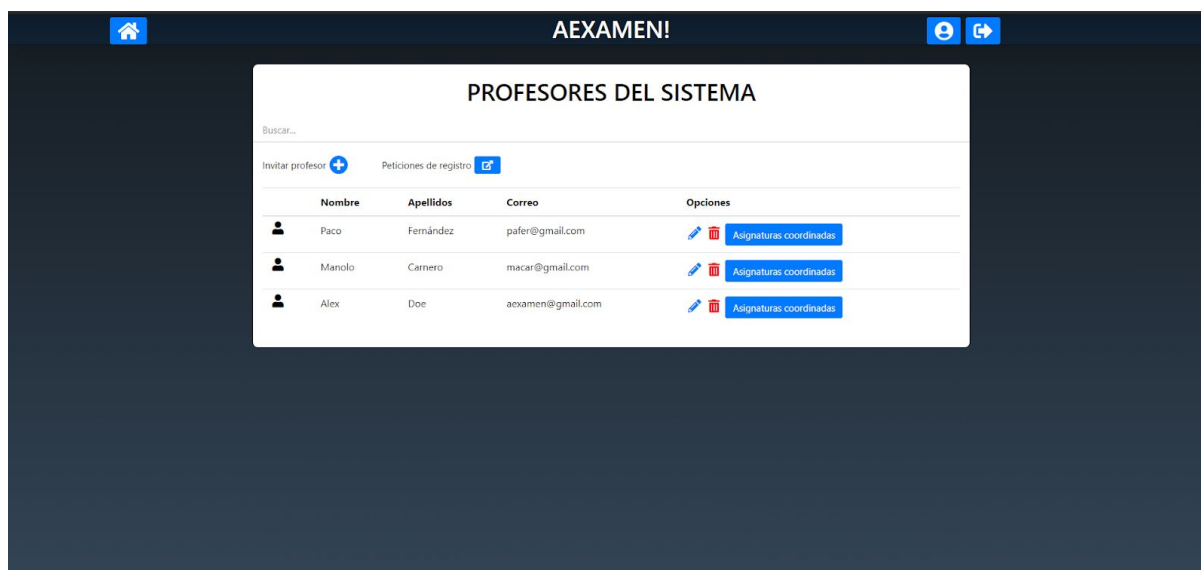


Figura 88. Profesores del sistema

### 3.5 Panel de control

En esta pantalla (Figura 89) el administrador podrá realizar acciones y ver el log y las peticiones de registro. Se puede descargar el “log” o verlo de manera online. También se puede borrar y reiniciar el log cuando este tenga demasiada información que no se quiera. También se muestran las peticiones de registro de personas que quiere acceder al sistema. Si se pulsa sobre el botón con “i” se muestra un mensaje que la persona ha escrito al enviar la solicitud. En caso de que se quiera aceptar la solicitud se pulsa sobre el botón con un tick y en caso de no querer aceptarla se pulsa sobre el botón con una “x”.

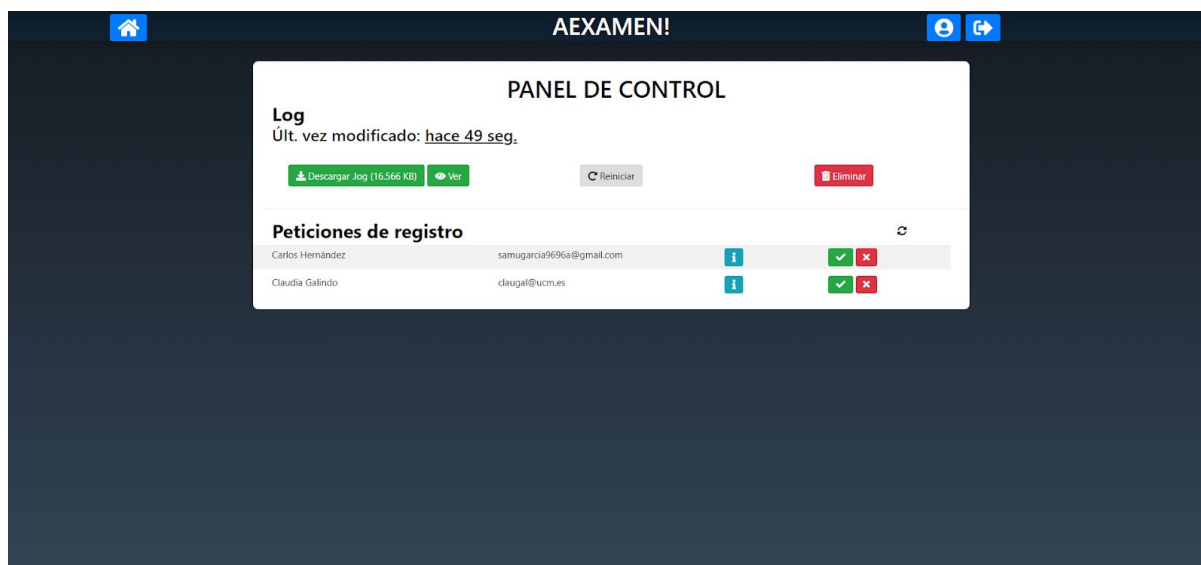
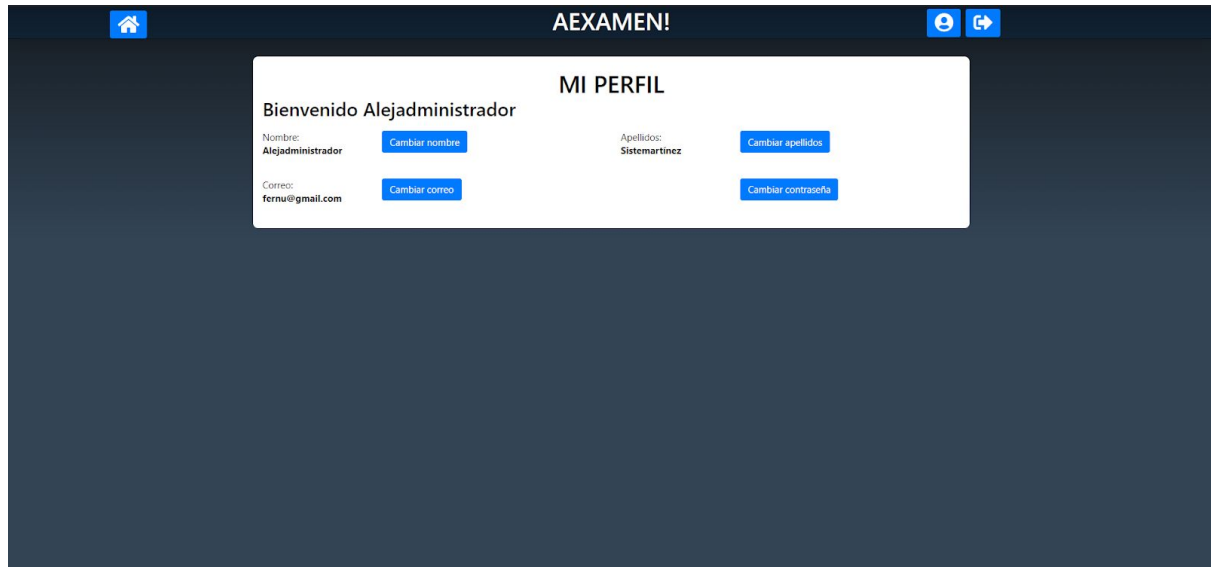


Figura 89. Panel de control

### 3.6 Perfil propio

En esta pantalla (Figura 90) el administrador puede hacer las mismas acciones que un profesor. Pero además podrá modificar su correo electrónico.



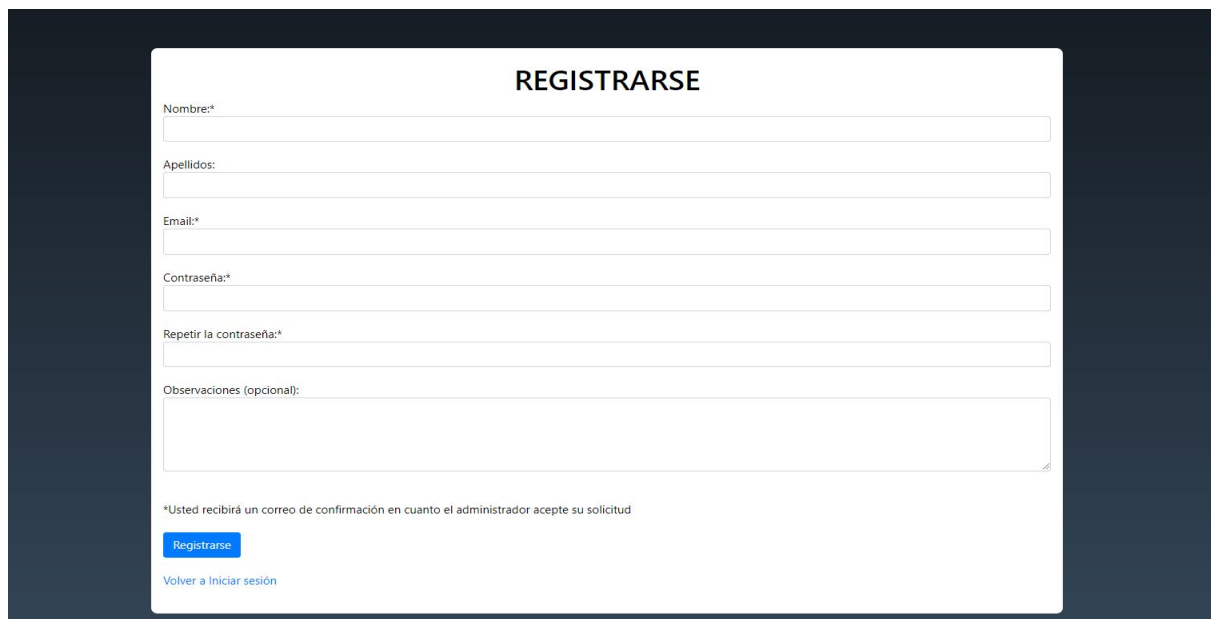
The screenshot shows a web interface for an administrator's profile. At the top, there's a dark blue navigation bar with a home icon, the text "AEXAMEN!", and two user icons. Below this, a white card titled "MI PERFIL" contains the text "Bienvenido Alejadministrador". Inside the card, there are three rows of information: "Nombre: Alejadministrador" with a "Cambiar nombre" button, "Apellidos: Sistemartinez" with a "Cambiar apellidos" button, and "Correo: fernu@gmail.com" with a "Cambiar correo" button. A "Cambiar contraseña" button is also present at the bottom right of the card.

Figura 90. Perfil propio administrador

## 4 Usuario no registrado

### 4.1 Registrarse

En la siguiente vista (Figura 91) el usuario debe rellenar todos los campos para enviar su solicitud de registro al sistema. Se incluye un campo de observaciones (opcional) para que el usuario pueda hacer alguna aclaración si lo ve necesario para que lo vea el administrador.



The screenshot shows a registration form titled "REGISTRARSE". It features several input fields: "Nombre:\*", "Apellidos:", "Email:\*", "Contraseña:\*", and "Repetir la contraseña:\*". There is also an optional text area for "Observaciones (opcional):". At the bottom, a note states: "\*Usted recibirá un correo de confirmación en cuanto el administrador acepte su solicitud". Below this note are two buttons: "Registrarse" and "Volver a Iniciar sesión".

Figura 91. Registrarse

## Anexo II: Guía de instalación de la aplicación

Para realizar la instalación de la aplicación se llevarán a cabo los siguientes pasos:

- Instalar la herramienta XAMPP, para lo cual se descarga el instalador desde la página oficial de XAMPP. A continuación se ejecuta el instalador (Figura 92).

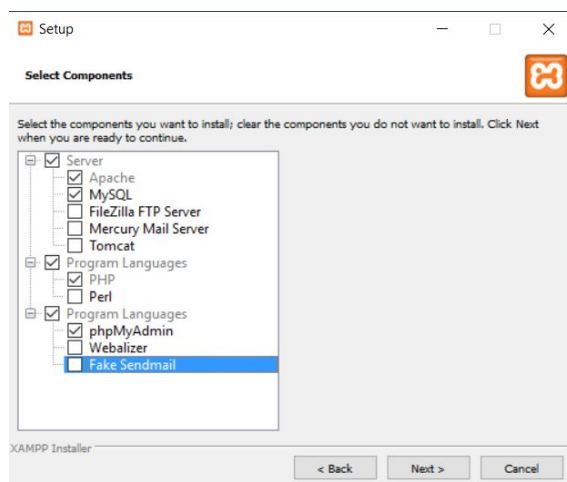


Figura 92. Instalación XAMPP (I)

Seleccionar para su instalación Apache, MySQL, PHP y phpMyAdmin y pulsar en siguiente. A continuación, se debe seleccionar una ruta para su instalación. Una vez finalizada su instalación, se inicia XAMPP. Aparecerá una pantalla en la cual se debe iniciar apache y MySQL pulsando los botones correspondientes (Figura 93).

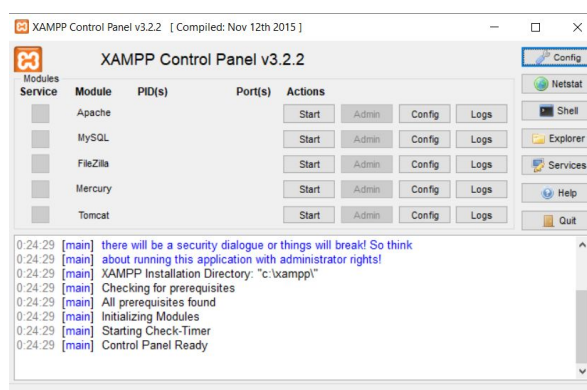


Figura 93. Instalación XAMPP (II)

- Se debe de definir la carpeta donde se situará el servidor apache en el fichero 'httpd.conf'. Este fichero se sitúa en '<ruta\_XAMPP>/apache/conf'. También se puede acceder a él pulsando en el botón config en el panel de control de XAMPP (Figura 94).

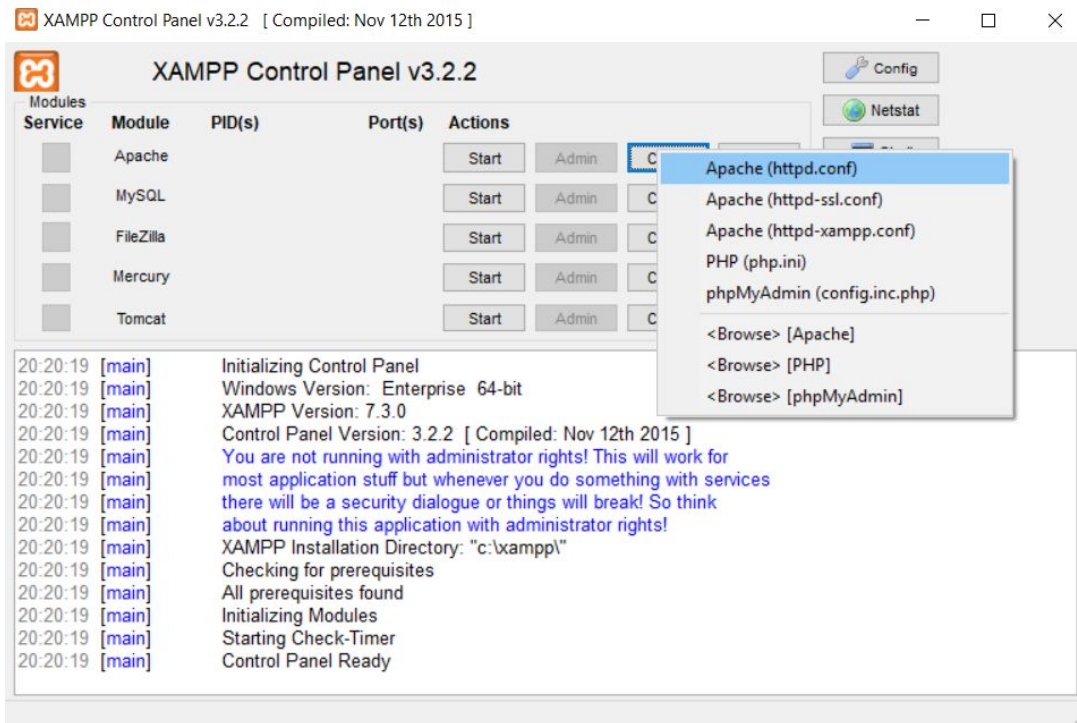


Figura 94. Acceso a archivo de configuración de Apache en XAMPP

- Una vez dentro del archivo (Figura 95) se tiene que buscar 'DocumentRoot' y definir la ruta que se quiere como raíz del servidor Apache. En esa carpeta es donde se va a situar el proyecto.

```

#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "C:/Users/Daniel/Documents/GitHub/aexamenCake/AExamen"
<Directory "C:/Users/Daniel/Documents/GitHub/aexamenCake/AExamen">
  
```

Figura 95. httpd.conf

- Entonces, se tendrá que copiar el código de la aplicación a la ruta que se haya establecido como directorio raíz del servidor Apache (Figura 96).

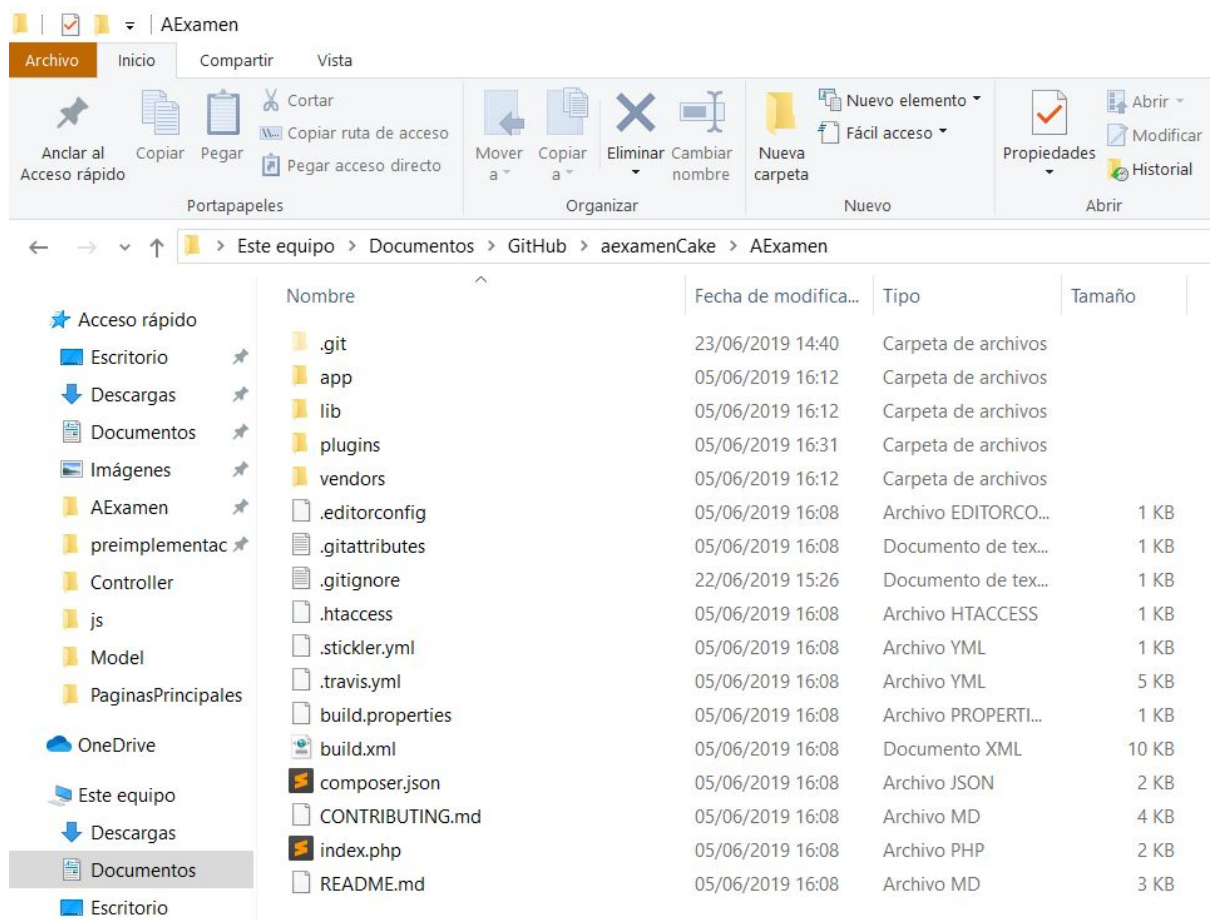


Figura 96. Directorio raíz del proyecto

- Acceder a la base de datos MySQL mediante PHPMYAdmin (Figura 97). Para ello en el navegador se indica la dirección “localhost/phpmyadmin”. En el archivo se crea un usuario para la base de datos con todos los permisos y que tenga el servidor en localhost. Para ello se pulsa sobre la opción “cuentas de usuario” y se rellenan los campos.

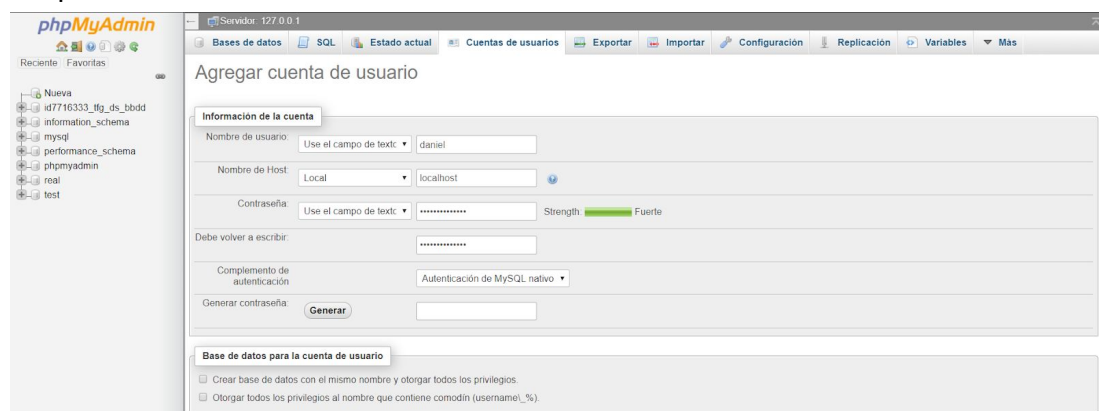


Figura 97. PHPMYAdmin (I)

- Importar el archivo donde está contenido la base de datos de la aplicación a la base de datos que está en local. Para ello se pulsa sobre la opción importar y luego se

pulsa sobre “seleccionar archivo” (Figura 98) y se selecciona el fichero .sql que contiene la base de datos de la aplicación.

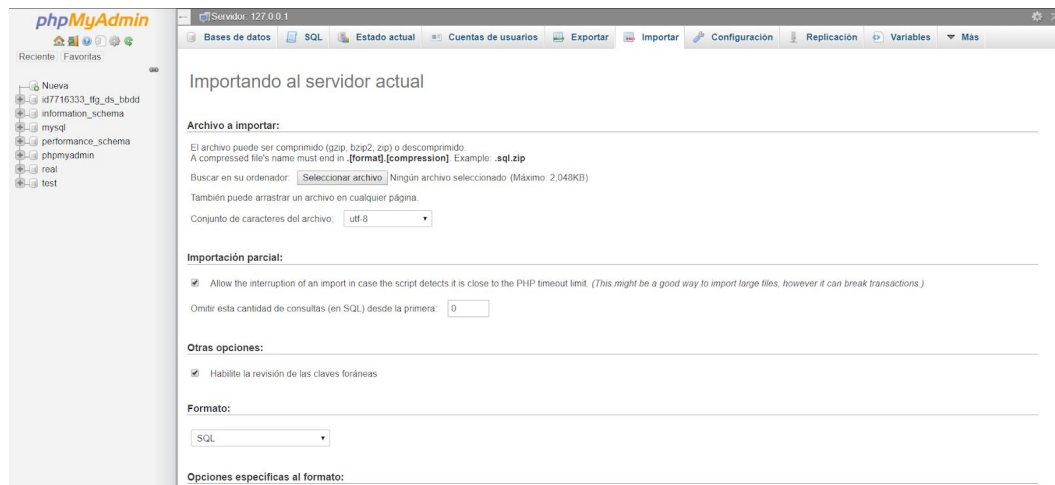


Figura 98. PHPMyAdmin (II)

- A continuación, se explicará cómo configurar el framework Cakephp para este proyecto:
  - a. Se clonará o descargar el proyecto y se situará en la carpeta raíz de nuestro servidor Apache
  - b. Se creará una carpeta llamada 'tmp' dentro de '/app'.
  - c. Se copiará el archivo 'database.php' en la carpeta /app/Config. Este fichero se proporcionará junto a la memoria.
  - d. Dentro de este fichero se definirán los valores de la base de datos, que se haya creado anteriormente (Figura 99).

```
public $default = array(
    'datasource' => 'Database/Mysql',
    'persistent' => false,
    'host' => 'localhost:3308',
    'login' => 'samdan',
    'password' => 'hola1234',
    'database' => 'aexamen',
    'prefix' => '',
    'encoding' => 'utf8',
);
```

Figura 99. /app/Config/database.php Valores por defecto de configuración de la base de datos

- Copiar el fichero 'credentials.json', adjuntado en el proyecto, en la ruta /app/webroot/json del código de la aplicación, poniendo todos los datos de la base de datos local.
- A continuación, se accede la aplicación desde el navegador web. Para ello se abre el navegador y se pone en la barra de búsqueda "localhost". De esta forma se muestra la pantalla de login de la aplicación (Figura 100). Los usuarios con las respectivas contraseñas se encuentran en el fichero "readme.txt" adjuntado en la entrega.



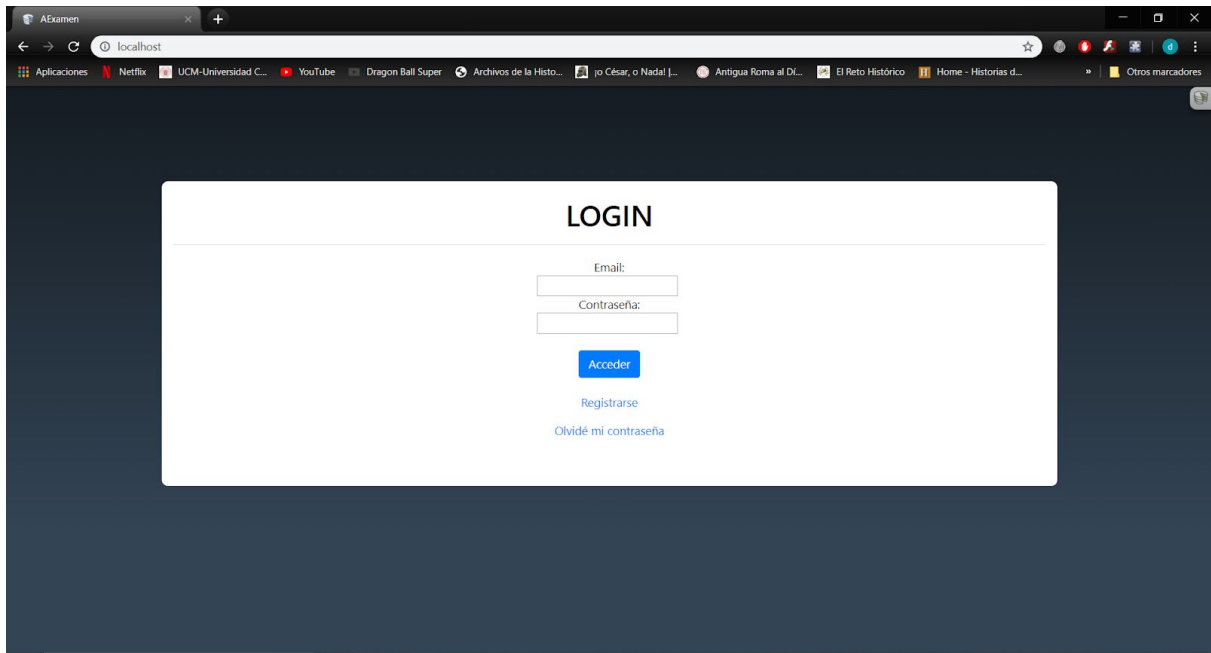


Figura 100. Pantalla inicial de la aplicación web